



DESIGN AND IMPLEMENTATION OF IMPROVED CENTRAL LOAD BALANCING ALGORITHM IN CLOUD ENVIRONMENT

Simranjeet kaur

Department of Computer Science and Technology,
ACET, Amritsar, India

Tejinder Sharma

Department of Information Technology,
ACET, Amritsar, India

Abstract: With the changes and advancement in technology Cloud Computing is playing a pivotal role in field of Technology with a remarkable growth. Through this paper a design and implementation of central load balancer is introduced with an effective results based upon various parameters. The proposal of the central balancer to balance the load on the cloud is introduced and weigh against existing load balancing methods. The foremost intention of these efforts is to initiate as well as appraise the projected load balancing algorithm by taking into account, the job duration of all solicited situation, the potential of every virtual machine (VM), and the correlation of numerous tasks. Rendition of the probable algorithm is deliberated besides contrasting with the existing methods.

Keywords: Cloud Computing , Load Balancing , ACO, Central Load Balancer.

I. INTRODUCTION

Cloud Computing is a class of collateral and administered system composed of a group of interlinked and implicit computers that are lively conferred as solitary or further unified computing amenity dependent on service positioned accords initiated through debate among the provider and consumers. It is an on-need network ingress to a apportion resource against earthly dispensed systems worldwide over internet, which is a consequences of minimal management attempt or service provider interaction. Furthermore, it is a variety of server-based computer where each and every application and evidence are stored on the web, the software programs will not run in the consumers PC, but rather accumulates and hasten on the server which can be attainable via Internet. This can be simply said to be "Deed Centric not PC centric" [1].

configurable computing -amenity (e.g., networks, servers, storage, applications, and services) that can be swiftly delivered with minimal management attempt or service giver interaction. This CLOUD prototype consists of five vital attributes, the service models trilogy (Software as a

Service / Platform as a Service / Infrastructure as a Service), and quartet deployment models involve private, community, public and hybrid cloud, while the quintet properties are: on-demand self-service, broad network access, resource pooling, rapid elasticity, and uniform assistance. [2]

II. JOBS SCHEDULING

The practice of allocating scheme assets to various tasks with the help of operating system (OS) is known as Job scheduling. The prioritized work queues which are pending CPU time are handled by the processor, and it must discover which job must be carried out from which queue along with the total time to be owed for the job. To be very definite that every job is carried out fairly and on time this sort of scheduling is carried out [3]. Importance of scheduling Algorithms-Minimize Response Time, Maximize Throughput and Fairness.

III. NEED FOR LOAD BALANCING

The Load Balancing is a sole affair associated to cloud computing, the load can constitute a CPU capacity, memory, network load etc. It is required to dispense the load uniformly amidst the nodes in a matrix. This outcome is nimble and systematic performance of the system. Thereby it evades steady loading or under loading of nodes in a mesh. It is cleaved into duo classes Static Load Balancing and Dynamic Load Balancing [4].

A. Static Load balancing Algorithm

In such algorithm the traffic is splitted constantly amidst the servers. It necessitates a foregoing knowledge of system resources, so that the resolution of relocating of the load does not depend on the present state of system. Static algorithm is bonafide in the system which has little contrast in load [5].

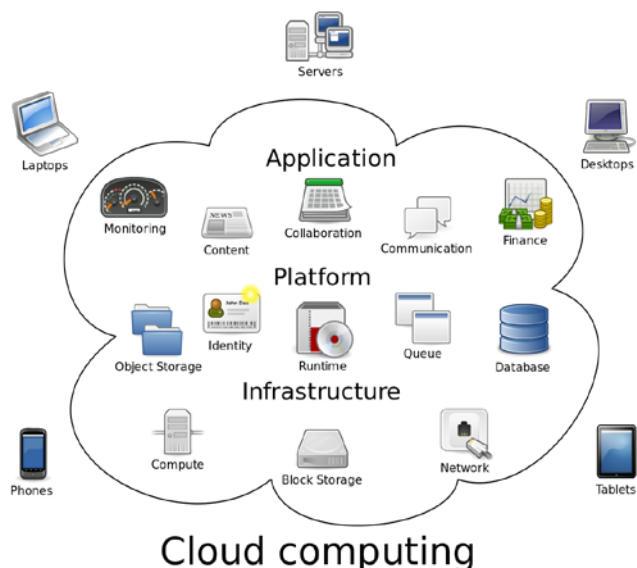


Figure 1. Cloud Model.

CLOUD computing is a replica for validating universal, appropriate, on demand grid ingress to a equity pool of

B. Dynamic Load balancing Algorithm

Such algorithm is the lightest server in the whole lattice or system is foraged and favoured for balancing .For this existent time transmission with tracery is required which can intensify the traffic in the structure. Here prevalent condition of the system is used to assemble resolution to direct the burden [5]. The framework upon which load balancing functions are [4]:

- Performance
- Throughput
- Fault Tolerance
- Migration Time
- Overhead
- Response Time
- Scalability

The principal focus of load balancing is to dispense the congestion amid the junction uniformly in the nest for the superior rendition of tracery [5]. The objective of load balancing is as proceeds:

- To intensify the surety of assistances to the consumer.
- To magnify the user satisfaction.
- To intensify usage of facility.
- To lessen the performance point and waiting phase of job imminent from unusual location.
- To make service rendered superior.
- Prolonged cluster firmness.
- Build a system that can tolerate the faults.
- Reconcile future modification.[5]

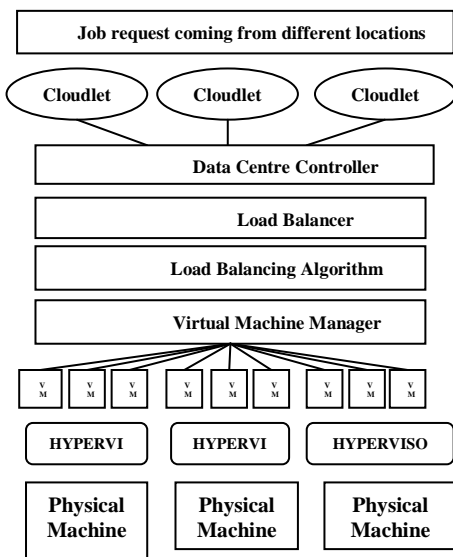


Figure 2. Structure of cloud computing environment [5]

In Cloud algorithms could be stratified like instant approach and batch means scheduling. In instant mode, scheduling organizes errands grounds on its onset by pertaining a least execution instance and lowest achievement time [6].

- **Round Robin:** It is a basic scheduling scheme that implies the notion of point in time slices. Accordingly, the point is alienated hooked on a variety of fragments, next every nodule is permitted towards employing a fastidious point interim. Within a time gap the nodule resolves its tasks performance. RR wields on casual choice of the virtual machines. The data centre controller allocates the tasks to the

accessible implicit machines deploying on a turning progression [6].

- **Weighted Round Robin:** It contemplates the abilities of the VMs as well as assigns superior amount of errands to the privileged facility VMs deployed on the supremacy specified to all of the VMs however it gets unsuccessful to consider the extent of the jobs to opt for the suitable VM [7].
- **Equally Spread Current Execution:** In this algorithm the balancer conserves a listing table of virtual machines plus their conditions (Available or Busy).To achieve the recommended job the client/server initially creates a plea to information centre to discover a proper (VM). The data centre enquire the load balancer used for allowance of the machines. It also inspects the table from crest until the initial existing VM is set up otherwise the index table is scoured fully [8]
- **Min-Min:** This algorithm is heuristic; initially it searches the lowest prospect time of each and every task. Following this it determines the tasks by means of least implementation time amongst every task and allots particular task onto that reserve. Identical steps are continual by min-min algorithm unless the entire duties are not planned. The restraint of this algorithm is that it fixes the least performed duties firstly along with dwell in assets which include elevated calculation influence. The superior tasks wait until the minor gets processed [9].

IV. LITERATURE SURVEY

Because of the significance and complication in the cloud network of load balancing assorted researchers tried to discovered unique techniques for boosting load balancing.

In [6], the author tried to deal through recital issues in computing is balancing. In actuality, a well-organized load balancing grants to diminish the costs and maximizes the ease of usage of resources. Throughout, author studied Ant Colony Optimization Technique which has served the balancing process. In a succeeding stride, he proposes an enhanced load style and algorithm for Cloud computing which aims to permit a superior retort instance.

In [10], the author tried to overcome issues related with the load management, fault tolerance and different security issues in cloud environment. The load can be CPU load, memory capacity, delay or network load in this paper author proposed a method based on Ant Colony optimization to resolve the problem of load balancing in cloud environment.

All the way through [11], instigator bestowed a designed Central Load Balancing Policy for Virtual Machines (CLBVM) to stable the weight consistently in a dispersed computing environment.

The authors [12], defines a bolted multi-operational load balancing resolution that circumvents the utilization of splitted memory in variance to alternative multiprocessing load balancing programmer which employ allocated memory and hasp to conserve a user session.

In this research study [13], Author proposed the ACCLB to balance the load on the cloud and compared it with the existing load balancing methods such as Vector Dot, Join idle queue. The main objective will be to balance the load on cloud and to reduce the energy consumption as compared to previous, on the cloud by using proposed method. Also we have to prove that our proposed technique is more

efficient for load balancing and energy consumption on cloud.

One more creator [14], seeked out a distributed algorithm for computing compare and balance built for evaluating to outstretch the equipoise resolution. They designed a basic replica which contracts the relocation instance of virtual machines via mutual cache in addition to attain the nil-downtime rearrangement. Subsistence VM passage has mostly duo rendition concerns: 1) Complete migration span: It is the entire instant time to relocate virtual machines against the host machine to the aimed device. 2) Downward phase: In this phase, the period of time at which services are unavailable to the users.

A. Gaps in Literature

In ACO resolution rely on randomly generated series besides optimizing the random results the procedure is executed several times orderly to reach the convergence point. This will capture plenty of time. In exchange for conquering the raised problem, to stable the load amongst virtual machines in the data centre Central Load Balancer algorithm is proposed. For allocation of requests Data Centre Controller inquires the Central Load Balancer. It accepts next less high preferred virtual machine if the states of virtual machine are Busy. By conducting the analysis, it has been initiated that most of the existing work in the research has omitted as a minimum one of the following concepts: While majority of the prevailing processes are based upon Ant Colony Optimization, thus, it possibly will result in the creation of random solutions population. To obtain optimal results as well as the time taken to get the convergence point is large. Time and Cost would have struck accurately however; the parallel increment has weakened the existing methods [15].

B. Problem Faced

Load balancing in cloud is the course of distributing the job load amongst different nodes in a distributed system used for enhanced reserve consumption and job reply time. The load balancer calculates observed in the particular time-span along with uses this value to estimate the virtual machine availability for the next time span. Load balancing ensures that all the processor in the system or every node in the network does roughly the equal quantity of work at any instant of time. It is a procedure of passing on the entire load to each nodes of the combined structure to create resource utilization efficient as well to get better response time of the situation, concurrently removing a situation in which some of the nodes are over loaded while several others are under loaded. It can be observed CPU utilization, throughput etc. will be improvised while balancing the load to virtual machines on the basis of utilization of resources on an immediate time. The goal of this research work is to set aside resources with the intention that we can assign the assets to additional number of processes in order to boost productivity plus make it environmental friendly [15].

V. PREDICATED LOAD BALANCING SYSTEM

A. Simulation Agenda

However Clouds formulate deployment of bulky sized approaches facile and economical, moreover it generates novel affairs for planners. Because Cloud framework are strewn, entreaty can be positioned in dissimilar confined

locale, with the selected allotment of the supplication collision with its rendering for users there are extreme from the data centre.

Simulators like Cloud Sim can be used for consenting the power along with interactivity of trials. A simulation demonstration involves imitations of correlative requisitions and infrastructures. Both the target infrastructure as well as the software in a speech that is elucidated by the simulator necessitates a hardly any strains from appeal developers to replica simulations. They are devised to be competent in universal experiments, and so modelling of definite scenarios might be time challenging still simulators endorse to copy such states.

Solitary the foremost purpose of Cloud Analyst is to take apart the clone security work out from a programming implementation, thus a modeller can spotlight on the counterfeit convolutions lacking a good deal of time on the mechanics of programming by the mode of a clone toolkit. The Cloud Analyst moreover qualify a modeller to continually carry out simulations with demeanour and a sequence of simulation experiments by means of minor restrictive disparity in a rapid and effortless manner [8].

B. Proposed Algorithm

To compare and analyse the results of original state-of-the-art with the proposed algorithm on the basis of following metrics:

- Overall Response Time.
- Data Centre Service Time.
- Transfer Cost.
- Throughput

The steps of proposed algorithm are as follow [16]:

1) Current Resource Balancer conserves a record that comprises states (BUSY/AVAILABLE), virtual machine id (VM id), timestamp and primacy of VM with the onset, the entire Virtual Machines are in accessible state and possess timestamp 0.

2) Data Centre Manager encounters a novel demand.

3) Data Centre Manager enquires the Current Resource Load Balancer for subsequently allotment and increase the timestamp of those requests which are arriving.

4) CRLB dissect the workbench from crest to perceive the utmost precedence virtual machine, available state.

5) If found:

a) The Current Resource Load Balancer returns the id of that machine to the Data Centre Manager.

b) The Centre Controller inform the VM which is identified by VM id and at the same time Load Balancer will update the timestamp of arriving process and if timestamp increasing Load balancer will update the priority of process.

c) For the latest provision of assets DC Manager reveals the Load Balancer of.

d) Current Resource Load Balancer re-equips the list consistently.

6) If no more initiated:

a) The Current Resource LB recur-1.

b) The Data Centre Manager streams the petition and reduces its timestamp.

7) Data Centre Manager inherit the riposte cloudlet in addition to it apprises the existing Load Balancer of the VM

de-allowance once the VM terminates the actions of requests

8) The Data Centre Manager uphold whether within reach be any invocations in the queue. If there are, it perpetuates from step 3.

9) Either draws out from step 2.

C. Proposed Central Load Balancer

To give out the consignment among the slave processors this Algorithm operate vital node as an agenda. To assign the situation to the mainframe the one which have the lowest load operate to be followed to prefer for the slave processor. The innermost processor tries to accrue all slave processors load data and confine the conclusion according to the data for what user awaits from excellent recital while operating this algorithm.

The anticipated load balancing algorithm “Central Load Balancer” will equalize the load amidst virtual machines having dissimilar hardware configurations and will give out the burden depending on conditions of virtual machines and apparatus design in data centre. The projected skill would subsist and be capable to achieve rapid furthermore balancing in computing milieu during operation of every machine according to their abilities.

Each appeal against user bases approaches at Data Centre Controller, within this projected method. DCC enquires the Central Balancer in favour of allowance of needs. Central Load Balancer upholds a bench which consisted of ids, priority and conditions of virtual machines. Central Load balancer dissects the table and come across with the peak precedence VM’s followed by verification of its situations and whenever its states are accessible next revisit the same virtual machine id (VM id) heading towards Data Centre Controller. It selects the subsequently a smaller amount soaring precedent virtual machine if the state of machine is eventful.

To conclude Data Centre Controller dispenses the appeal to VM id that is provided by Central Load Balancer (CLB).The Central Load Balancer is allied toward each and every user along with virtual machines currently present in cloud data Centre with the help of Data Centre Controller .Based ahead the CPU speed (MIPS) and memory the Balancer reckon the primacy of virtual machines.

D. Virtual Machine Priority Calculation

The preference of every virtual machine is purposive constructed on its CPU speed (MIPS) and memory. It is calculated as underneath:

Priority can be calculated using the formulas below:

$$\text{Load } 1_{ij} = \frac{\sum_{i=1}^n \text{cloudlet_length}_{ij}}{\text{Vm}_j_mips} \quad (i)$$

where Vm_j_mips is defined by millions of instructions per second for each processor of Vm_j , n is the total no of scout foragers, $fit_{i,j}$ defines the load of machines (i) for Vm_j belongs to cloudlet number, cloudlet length is defined as the task length that has been submitted to Vm_j .

The virtual machine (Vm_j) capacity is being calculated using the following parameters:

$$\text{Capacity_Vm}_j = \text{Vm}_j_cpu * \text{Vm}_j_size + \text{Vm}_j_bandwidth \quad (ii)$$

Where Vm_j_cpu , the number of processors in a virtual machine Vm_j is Vm_j_size the virtual machine memory size,

$\text{Vm}_j_bandwidth$ is the network bandwidth ability of Virtual Machine Vm_j .

$$\text{Load } 2_{ij} = \frac{\sum_{i=1}^n \text{cloudlet_length}_{ij}}{\text{Vm}_j_Size} \dots\dots\dots (iii)$$

$$\text{VM Load} = \text{Mean of load1} + \text{Capacity_vm} + \text{load } 2 \dots\dots\dots (iv)$$

E. Performance Evaluation

In Cloud Analyst tool the anticipated algorithm is executed and united. This analyst tool is a Cloud Sim-deployed tool for representation and exploration of large cloud computing environment. We agreed our tryout based on all the parameters like: Response time, Data Centre Processing Time, Throughput, Transfer cost also user base similar to the most of the real-world net applications.

1) *Average Response Time and Data Centre (DC) Processing Time:* The response time for six user bases and three data centres is being simulated and executed by cloud analyst respectively, where average response time and data Centre processing time is calculated as:

$$\text{Response time} = (\text{Response}_{\text{Received } \Delta t} - \text{Request}_{\text{Sent } t})$$

$$\text{Response}_{\text{Received } \Delta t} = \frac{\sum \text{data_size}}{\sum_{c=1}^c \text{request}}$$

Where $\sum_{c=1}^c \text{request}$ is the total number of requests received by the virtual machine at time Δt . $\text{Request}_{\text{Sent } t}$ is the job request sent to the virtual machine at time t. $\sum \text{data_size}$ is the total data processed.

$$\text{DC_Processing_Time} = \text{DC_end_time} - \text{DC_start_time}$$

Where start time is the starting time when the data Centre is being started processing the task and is measured by `gridsim_clock`. End time is the ending time when data Centre is finished the serving of the allotted task and is measured by `gridsim_clock`.

2) *Processing Cost:* The Processing cost for six user bases and three data centers is being calculated by cloud analyst for each load balancing policy. Where Total Processing Cost is calculated as:

$$\text{TP}_{\text{cost}} = \text{DT}_{\text{cost}} + \text{VM}_{\text{cost}}$$

Where DT_{cost} is the total data transfer cost. VM_{cost} is the virtual machine cost.

$$\text{DT}_{\text{cost}} = \text{dataGB} * \text{costPerDataGB}$$

$$\text{dataGB} = \frac{\sum \text{data}}{(1024 * 1024)}$$

$$\text{VM}_{\text{cost}} = \sum \text{VM_usagetime} * \text{costperVmHour}$$

$$\sum \text{VM_usagetime} = \text{VM_time}_{t+1} - \text{VM_time}_t$$

Where VM_time_{t+1} is the time when the virtual machine is released after performing the task and VM_time_t is the time when the virtual machine is allocated to the user base

VI. RESULTS AND DISCUSION

Data center aim which consists no. of VM, storage, processor speed, VM image size, bandwidth, memory, VM policy should be define under data center configuration tab.

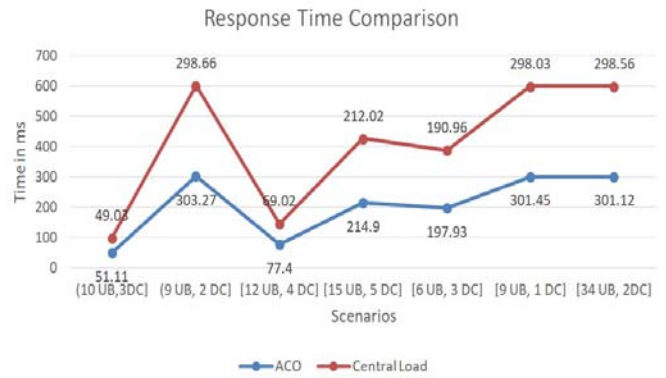
Table I. Parametric table[17]

Parameters	Value used
Request Per User Per Hour	60
Data Size Per Request	1000
Number of Region Use	2
Peak Hour Start(GMT)	3
Peak Hour End(GMT)	9
VM Image size	10000
VM Memory	512
VM Bandwidth	1000bps
DC1- Storage Per Machine	100000000 Mb
DC1- Available BW Per Machine	1000000
DC1- No Of Processors Per User	4
DC1- Processor Speed	10000 MIPS

Consider analysis case to investigate the effectiveness of central balancer. In former load is retained steady furthermore the figure of virtual machines are improved in addition to with the following case, amount of VM are stowed stable(10)along with the load is enlarged continually all the way through amended data size per request.

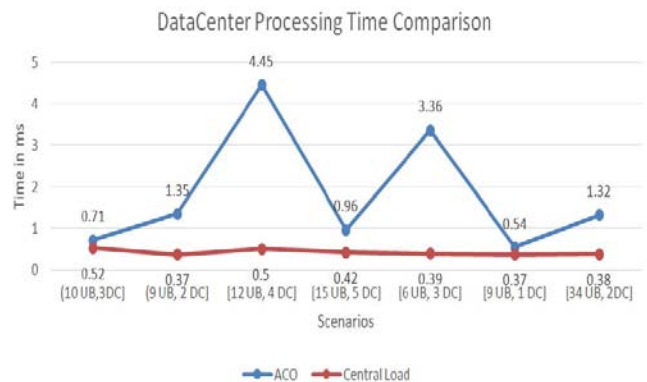
Case –I the number of virtual machines varies for response time.

Datasets	ACO[6]	Central Load
Scenario1(10UB,3DC)	51.11	49.03
Scenario2(9UB,2DC)	303.27	298.66
Scenario3(12UB,4DC)	77.4	69.02
Scenario4(15UB,5DC)	214.9	212.02
Scenario5(6UB,3DC)	197.93	190.96
Scenario6(9UB,1DC)	301.45	298.03
Scenario7(34UB,2DC)	301.12	298.56



Case –II The quantity of virtual machines varies for Data Centre Processing Time Comparison.

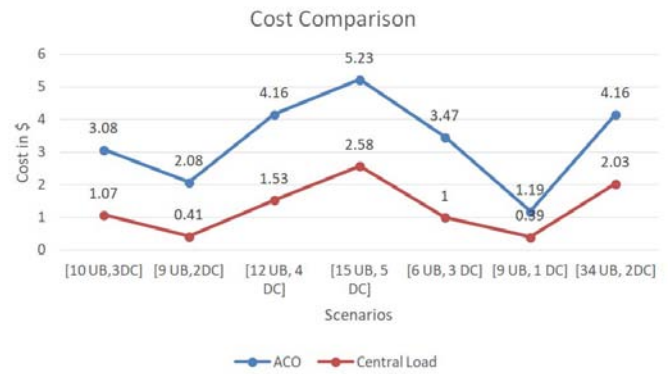
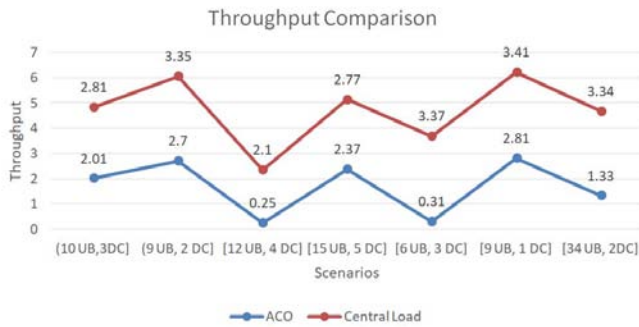
Datasets	ACO[6]	Central Load
Scenario1(10UB,3DC)	0.71	0.52
Scenario2(9UB,2DC)	1.35	0.37
Scenario3(12UB,4DC)	4.45	0.5
Scenario4(15UB,5DC)	0.96	0.42
Scenario5(6UB,3DC)	3.36	0.39
Scenario6(9UB,1DC)	0.54	0.37
Scenario7(34UB,2DC)	1.32	0.38



Case –III The figure of virtual machines differ for throughput.

Datasets	ACO[6]	Central Load
Scenario1(10UB,3DC)	2.01	2.81
Scenario2(9UB,2DC)	2.7	3.35
Scenario3(12UB,4DC)	0.25	2.1
Scenario4(15UB,5DC)	2.37	2.77
Scenario5(6UB,3DC)	0.31	3.37
Scenario6(9UB,1DC)	2.81	3.41

Scenario7(34UB,2DC)	1.33	3.34
---------------------	------	------



Case –IV The figure of virtual machines differ for cost.

Datasets	ACO[6]	Central Load
Scenario1(10UB,3DC)	0.71	0.52
Scenario2(9UB,2DC)	1.35	0.37
Scenario3(12UB,4DC)	4.45	0.5
Scenario4(15UB,5DC)	0.96	0.42
Scenario5(6UB,3DC)	3.47	1
Scenario6(9UB,1DC)	1.19	0.39
Scenario7(34UB,2DC)	4.16	2.03

VII. CONCLUSION AND FUTURE SCOPE

Within this planned Central Load Balancer (CLB) approach, the avoidance of conditions akin to over loading and under loading of implicit machines are taken in the concern. The Central Load Balancer (CLB) conduct load allocation amongst assorted virtual machines also designates the load consequently in the direction of their priority and situations. Within this fashion the method proficiently split the load of client needs between varieties of implicit machines. The potential of my efforts is to expand my concept with the aim to apportion the load to virtual machines according to their present reserve consumption such as processors current use as well as of the memory so the circulation of load would spirit up with more dynamic and robust balancing,

VIII. ACKNOWLEDGMENT

With lot zeal and in all modesty, I obliged my earnest gratitude to Amritsar College Of Engineering And Technology, CSE Department, Amritsar India intended for the constant support, liberal assistance and estimated supervision all through the phase of this study. The support of adviser is immensely valuable to take away the exploration in this field.

Table II. Comparison of proposed with existing algorithm

Datasets ACO and Central load	Response time		Datacenters Processing Time		Throughput		Cost	
	ACO[6]	Central	ACO[6]	Central	ACO[6]	Central	ACO[6]	Central
6UB,3DC	197.93	190.96	3.36	0.39	0.31	3.37	3.47	1
9UB,1DC	301.45	298.03	0.54	0.37	2.81	3.41	1.19	0.39
9UB, 2 DC	303.27	298.66	1.35	0.37	2.7	3.35	2.08	0.41
10UB,3DC	51.11	49.03	0.71	0.52	2.01	2.81	3.08	1.07
12UB,4DC	77.4	69.02	4.45	0.5	0.25	2.1	4.16	1.53
15UB,5DC	214.9	212.02	0.96	0.42	2.37	2.77	5.23	2.58
34UB,2DC	301.12	298.56	1.32	0.38	1.33	3.34	4.16	2.03

IX. REFERENCES

[1] M. Miller, "Cloud Computing Web-Based Applications That Change the Way You Work and Collaborate", Que Publishing, First Edition, 2008.

[2] L.Schubert "Advances in clouds research in Future CloudComputing",Public Version1.0, report January 2011.

[3] <httpswww.techopedia.comdefinition7882job-scheduling>.

- [4] Uma, J., V. Ramasamy, and A. Kaleeswaran. "Load Balancing Algorithms in Cloud Computing Environment-A Methodical Comparison ." International Journal of Engineering Research and Technology 3, no. 2 .pp.272-275, (2014).
- [5] Gupta, Shikha, and S. Sanghwan. "Load balancing in cloud computing: A review." International Journal of Science, Engineering and Technology Research (IJSETR) 4, Vol no. 6 (2015).
- [6] Ragmani, A., El Omri, A., Abghour, N., Moussaid, K. and Rida, M., "A performed load balancing algorithm for public Cloud computing using ant colony optimization". In Cloud Computing Technologies and Applications (CloudTech), 2nd International Conference (PP. 221-228),May 2016 IEEE.
- [7] Devi, D. Chitra, and V. Rhymend Uthariaraj. "Load balancing in cloud computing environment using improved weighted round robin algorithm for nonpreemptive dependent tasks." The scientific world journal (2016).
- [8] Wickremasinghe, B., Calheiros, R.N. and Buyya, R., "Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications". In Advanced Information Networking and Applications (AINA), PP.446-452. April 2010, IEEE.
- [9] Kaur, R. and Patra, P.K. "Resource Allocation with improved Min Min Algorithm". International Journal of Computer Applications, Vol.76-(15), PP.0975-8887, August,2013.
- [10] Mishra, R. and Jaiswal, A., "Ant colony optimization: A solution of load balancing in cloud". International Journal of Web & Semantic Technology, Vol.3(2), PP.33, April 2012.
- [11] Bhadani, A. and Chaudhary, S. "Performance evaluation of web servers using central load balancing policy over virtual machines on cloud". In Proceedings of the Third Annual ACM Bangalore Conference, PP. 16. ACM, January 2010
- [12] Liu, X., Pan, L., Wang, C.J. and Xie, J.Y. "A lock-free solution for load balancing in multi-core environment". In Intelligent Systems and Applications (ISA),3rd International Workshop PP. 1-4, may 2011, IEEE.
- [13] Kaur, G. and Kamboj, S, "A REVIEW ON BALANCING THE LOAD ON CLOUD USING ACCLB HYBRID LOAD BALANCING TECHNIQUE". In International Journal of Technology and Computing (IJTC) ,Vol. 2, No. 7 (July, 2016).
- [14] Wickremasinghe,B. and Buyya, R., "CloudAnalyst: A CloudSim-based tool for modelling and analysis of large scale cloud computing environ-ments". MEDC project report, Vol.22(6), pp.433-659.2009.
- [15] Efficient Load Balancing using Improved Central Load Balancing Technique.
- [16] Soni, G. and Kalra, M, "A novel approach for load balancing in cloud data center". In Advance Computing Conference (IACC), International (pp. 807-812), 2014, February IEEE.
- [17] Patel, S., Patel, R., Patel, H. and Vahora, S. "CloudAnalyst: A Survey of Load Balancing Policies". International Journal of Computer Applications, Vol.117(21),2015.