e-ISSN: 2349-9745
p-ISSN: 2393-8161

Scientific Journal Impact Factor (SJIF): 1.711

International Journal of Modern Trends in Engineering and Research

www.ijmter.com

Efficient Throttled Load Balancing Algorithm in Cloud Environment

Durgesh Patel[1] , Mr. Anand S Rajawat[2]

[1] *P. G. Scholar, Department of CSE, SVITS Indore, M.P., India*

[2]*Assistant Professor, Department of CSE, SVITS Indore, M.P., India*

Abstract - This paper presents an approach for scheduling algorithms that can maintain the load balancing and provides better improved strategies through efficient job scheduling and modified resource allocation techniques. In modern days cloud computing is one of the biggest environment platform which provides largest storage of more data in very minimum cost and available for all time on the internet. But the cloud computing environment has more critical issue like security, load balancing and fault tolerance ability. In this paper we are focusing on Load Balancing approach. The Load balancing is the process of distributed load on the various nodes which provides best resource utilization when nodes are overload with job. Load balancer is required to handle the load when one node is overloaded. When the node is overloaded at which time that loads is distribute on another free node.

We have proposed Throttled scheduling algorithm and compared it with the round robin, ESCE and Throttled scheduling to estimate response time, processing time, which is having an impact on cost.

Keywords- Cloud Computing, Simulation, Virtualization, Round Robin and ESCE Algorithms, Throttled Algorithms, Load Balancing.

1. INTRODUCTION

Cloud computing provide infrastructure, platform, and software as services. These services are using pay-as-you-use model to customers, regardless of their location. Cloud computing is a cost effective model for provisioning services and it makes IT management easier and more responsive to the Changing needs of the business [1]. The access to the infrastructure incurs payments in real currency in cloud environment. Today network bandwidth, Less response time, minimum delay in data transfer and minimum data transfer cost are main challenging issues in cloud computing load balancing environment .In this study based on clouds scheduling algorithms round robin, ESCE (Equally Spread Current Execution) algorithms and compare them. Cloud analyst is simulation based approach. The Simulation based approaches provide significant Benefits, as it allows researchers to test their proposed algorithms and protocols in a repeatable and controlled environment, and to find solution to the performance before deploying in the real cloud.

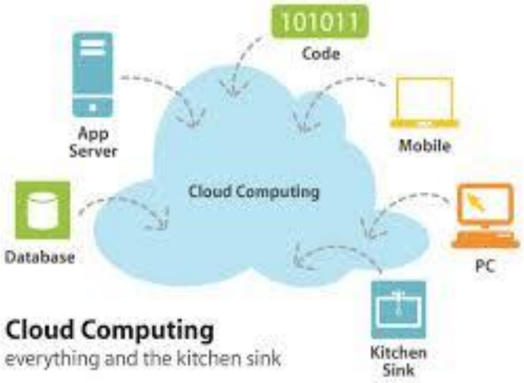


Figure 1: Cloud Computing Environment

Now a days, cloud computing is the heart favourite topic to many researchers. It will become more popular in coming years as the reach of internet is increasing day by day.
Cloud computing has three basic models, which are Platform as a Service (PaaS), Infrastructure or Hardware as a Service (IaaS/Haas), Software as a Service (SaaS).

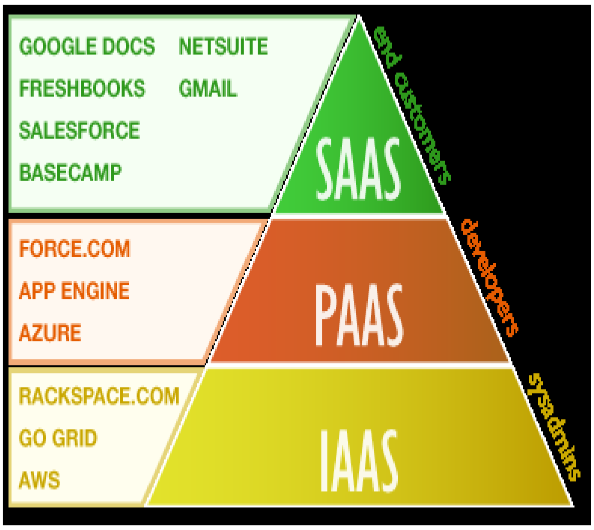


Figure 2: Cloud Architecture

The Main advantages of cloud computing are: low cost, improved performance, infinite storage space etc.
Load balancing in cloud computing systems is really a challenge now. A distributed solution is required. As it is not always practically feasible or cost efficient to maintain one or more idle services just as to fulfill the required demands. All jobs can't be assigned to appropriate servers and clients individually for efficient load balancing as cloud is a very complex structure and components are present throughout a wide spread area.

2. PROBLEM FORMULATION

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. 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.

3. EXISTING LOAD BALANCING ALGORITHMS FOR CLOUD COMPUTING

Distribute workload of multiple network links to achieve maximum throughput, minimize response time and to avoid overloading. We use three algorithms to distribute the load. And check the performance time and cost.

A. Round Robin Algorithm (RR):

It is the simplest algorithm that uses the concept of time quantum or slices Here the time is divided into multiple slices and each node is given a particular time quantum or time interval and in this quantum the node will perform its operations. The resources of the service provider are provided to the client on the basis of this time quantum. In Round Robin Scheduling the time quantum play a very important role for scheduling, because if time quantum is very large then Round Robin Scheduling Algorithm is same as the FCFS Scheduling. If the time quantum is extremely too small then Round Robin Scheduling is called as Processor Sharing Algorithm and number of context switches is very high. It selects the load on random basis and leads to the condition where some nodes are heavily loaded and some are lightly loaded. Though the algorithm is very simple but there is an additional load on the scheduler to decide the size of quantum [3] and it has longer average waiting time, higher context switches higher turnaround time and low throughput.

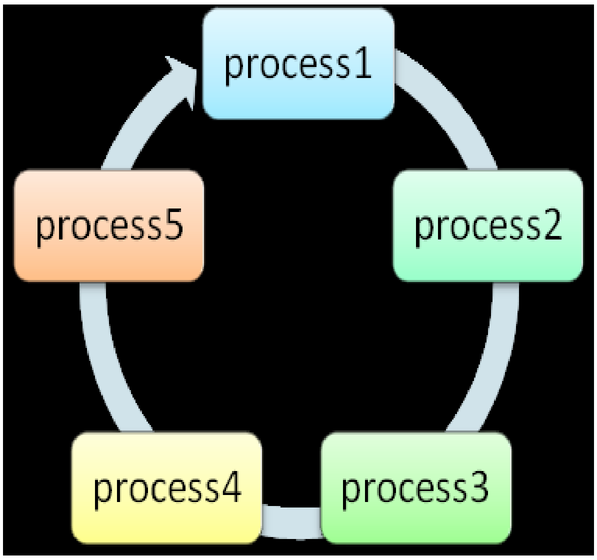


Figure 3: Round Robin Algorithm

B. Equally Spread Current Execution Algorithm (ESCE):

In spread spectrum technique load balancer makes effort to preserve equal load to all the virtual machines connected with the data centre. Load balancer maintains an index table of Virtual machines as well as number of requests currently assigned to the Virtual Machine (VM). If the request comes from the data centre to allocate the new VM, it scans the index table for least loaded VM. In case there are more than one VM is found than first identified VM is selected for handling the request of the client/node, the load balancer also returns the VM id to the data centre controller. The data centre communicates the request to the VM identified by that id. The data centre revises the index table by increasing the allocation count of identified VM. When VM completes the assigned task, a request is communicated to data centre which is further notified by the load balancer. The load balancer again revises the index table by decreasing the allocation count for identified VM by one but there is an additional computation overhead to scan the queue again and again

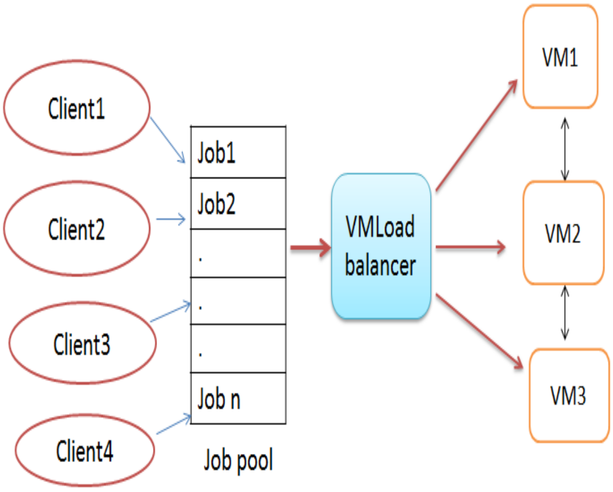


Figure 4: ESCE Algorithm

C. Throttled Load Balancing Algorithm (TLB):

In this algorithm the load balancer maintains an index table of virtual machines as well as their states (Available or Busy). The client/server first makes a request to data centre to find a suitable virtual machine (VM) to perform the recommended job. The data centre queries the load balancer for allocation of the VM. The load balancer scans the index table from top until the first available VM is found or the index table is scanned fully. If the VM is found, the load data centre.

The data centre communicates the request to the VM identified by the id. Further, the data centre acknowledges the load balancer of the new allocation and the data centre revises the index table accordingly.

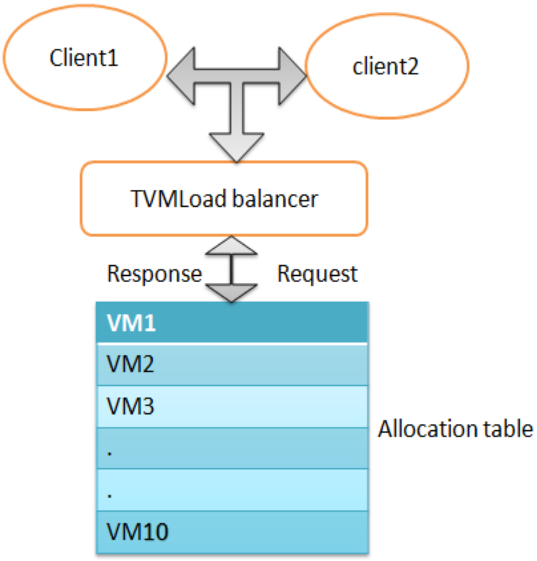


Figure 5: Throttled Algorithm

While processing the request of client, if appropriate VM is not found, the load balancer returns -1 to the data centre. The data centre queues the request with it. When the VM completes the allocated task, a request is acknowledged to data centre, which is further apprised to load balancer to de- allocate the same VM whose id is already communicated. The total execution time is estimated in three phases. In the first phase the formation of the virtual machines and they will be idle waiting for the scheduler to schedule the jobs in the queue, once jobs are allocated, the virtual machines in the cloud will start processing, which is the second phase, and finally in the third phase the cleanup or the destruction of the virtual machines. The throughput of the computing model can be estimated as the total number of jobs executed within a time span without considering the virtual machine formation time and destruction time The proposed algorithm will improve the performance by providing the resources on demand, resulting in increased number of job executions and thus reducing the rejection in the number of jobs submitted.

4. PROPOSED MODEL

Cloud computing is generally referred as the on demand service. In cloud computing infrastructure, software, and services are provided on demand at a specific time & also according to the user’s requirement. The load balancing algorithms are used for allocating correct virtual machine. The load balancing algorithm decides which VM is to be allocated against a user requirement. Therefore, the load balancing algorithms play a vital role in improving the cloud performance & for the overall resource utilization. The load balancing algorithm may be static or dynamic. Here, we are proposing a generalized model for cloud load balancing. It includes three load balancing algorithms: Round Robin, ESCE and Throttled. The proposed model is as follows:

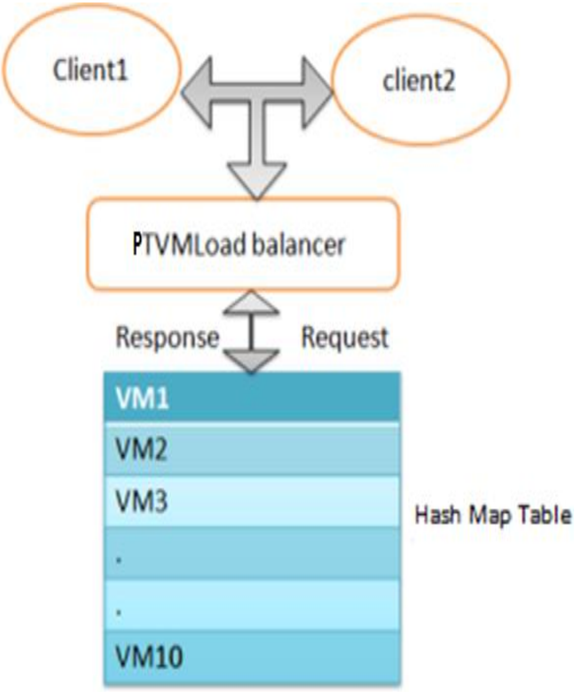


Figure 7: Efficient Throttled Algorithm

Proposed Algorithm:

Input:

- Data centre requests r1,r2,r3,r4,..., rn
- Available virtual machines vm1, vm2, vm3, vm4… vmn.

Output:

- Data centre requests r1,r2,……,rn are allocated available virtual machines vm1,vm2,………,vmn

Process:

1. The efficient throttled algorithm maintains a hash map table of all the available virtual machines which their current state and the expected response time. This state may be available or busy. At the beginning, all the virtual machines are available.
2. When data centre controller receives a request then it forwards that request to the efficient throttled load balancer. The efficient throttled load balancer is responsible for the virtual machine allocation. So that the job can be accomplished.
3. The efficient throttled algorithm scans the hash map table. It checks the status of the available virtual machine.

3.1 If a virtual machine with least load and the minimum response time is found.

- Then the efficient throttled algorithm sends the VM id of that machine to the data centre controller.
- Data centre controller sends a request to that virtual machine.
- Data centre controller sends a notification of this new allocation to the updated throttled.
- The efficient throttled algorithm updates the hash map index accordingly.

3.2 If a virtual machine is not found then the efficient throttled algorithm returns -1 to the data centre controller.

4. When the virtual machine finishes the request.

- The data centre controller sends a notification to efficient throttled that the vm id has finished the request.
- Efficient throttled modifies the hash map table accordingly.

If there are more requests then the data centre controller repeats step 3 for other virtual machines until the size of the hash map table is reached. Also of the size of hash map table is reached then the parsing starts with the first hash map index.

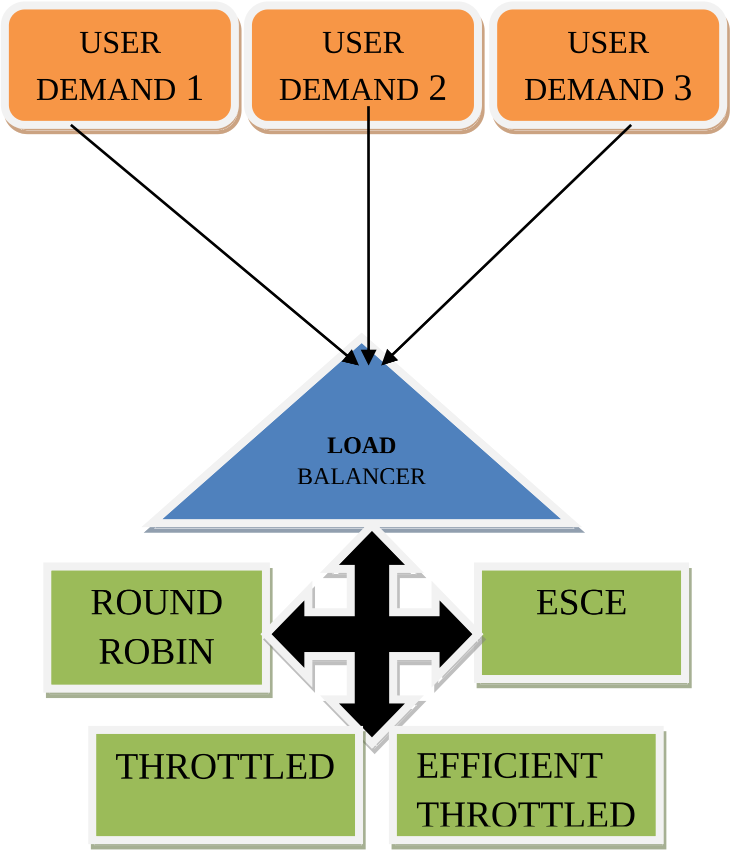


Figure 6: The Load Balancing Model

5. CLOUD ANALYST

Cloud Analyst [8] [9] [10] [11] is a GUI based tool that is developed on CloudSim architecture. CloudSim is a toolkit that allows doing modeling, simulation and other experimentation. The main problem with CloudSim is that all the work need to be done programmatically. It allows the user to do repeated simulations with slight change in parameters very easily and quickly. The cloud analyst allows setting location of users that are generating the application and also the location of the data centers. In this various configuration parameters can be set like number of users, number of request generated per user per hour , number of virtual machines, number of processors, amount of storage, network bandwidth and other necessary parameters. Based on the parameters the tool computes the simulation result and shows them in graphical form. The result includes response time, processing time, cost etc..

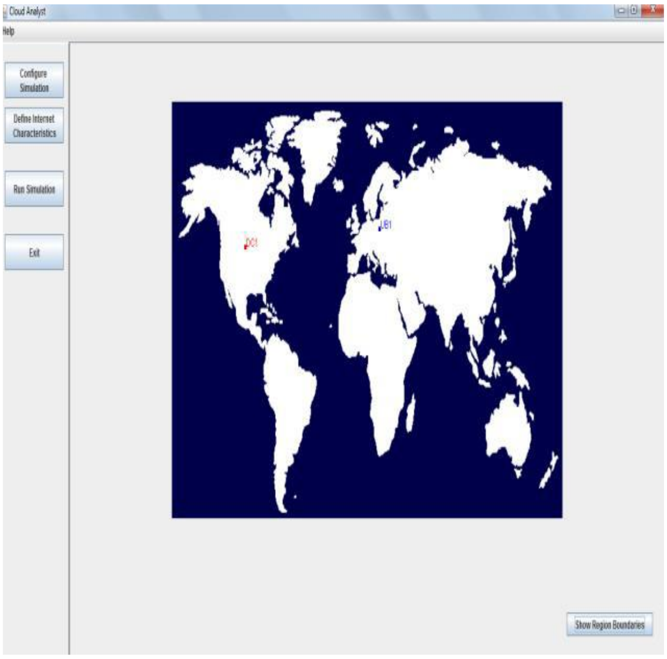


Figure 8: GUI interface

By performing various simulations operation the cloud provider can determine the best way to allocate resources, based on request which data center to be selected and can optimize cost for providing services

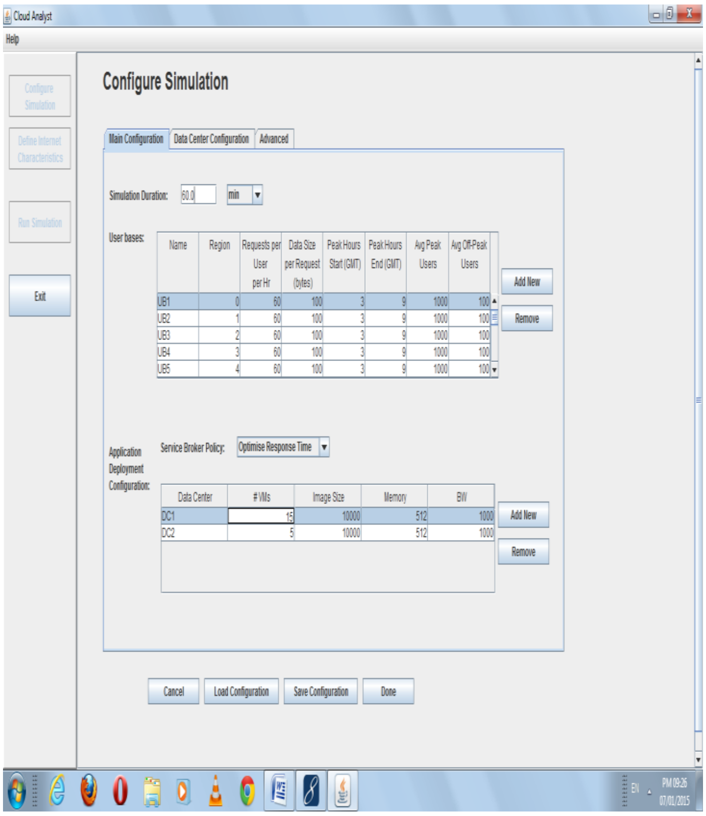
5.1 Simulation Parameters:

5.1.1 Region

In the Cloud Analyst the world is divided in to 6 ‘Regions’ that coincide with the 6 main continents in the World. The other main entities such as User Bases and Data Centers belong to one of these regions. This geographical grouping is used to maintain a level of realistic simplicity for the large scaled simulation being attempted in the Cloud Analyst.

5.1.2 Users

A User Base models a group of users that is let as a single unit in the simulation process and its main responsibility is to create traffic for the simulation process.



Configure Simulation

Simulation Duration: 60.0 min

User bases:

Name	Region	Requests per User per Hr	Data Size per Request (bytes)	Peak Hours Start (GMT)	Peak Hours End (GMT)	Avg Peak Users	Avg Off-Peak Users
UB1	0	60	100	3	9	1000	100
UB2	1	60	100	3	9	1000	100
UB3	2	60	100	3	9	1000	100
UB4	3	60	100	3	9	1000	100
UB5	4	60	100	3	9	1000	100

Application Deployment Configuration:

Service Broker Policy: Optimise Response Time

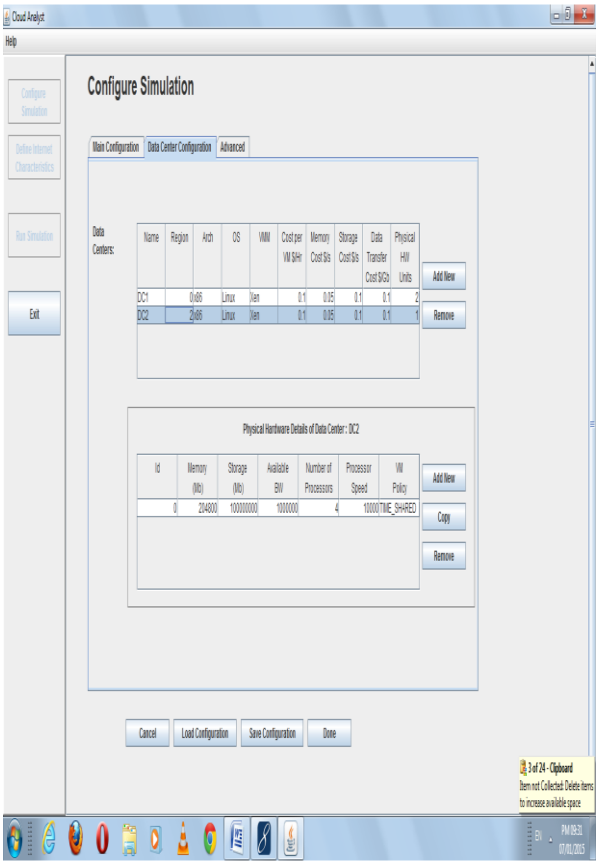
Data Center	# VMs	Image Size	Memory	BW
DC1	15	10000	512	1000
DC2	5	10000	512	1000

Figure 9

A single User Base may represent thousands of users but is configured as a single unit and the traffic generated in simultaneous bursts representative of the size of the user base. The modeller may choose to use a User Base to represent a single user, but ideally a User Base should be used to represent a larger number of users for the efficiency of simulation.

5.1.3 DataCenterController

The Data Center Controller is probably the most important entity in the Cloud Analyst. A single Data Center Controller is mapped to a single cloudsim. Data Center object and manages the data center management activities such as VM creation and destruction and does the routing of user requests received from User Bases via the Internet to the VMs. It can also be viewed as the façade used by Cloud Analyst to access the heart of CloudSim toolkit functionality.



Configure Simulation

Data Centers:

Name	Region	Arch	OS	VMM	Cost per VM $/Hr	Memory Cost $/s	Storage Cost $/s	Data Transfer Cost $/Gb	Physical HW Units
DC1	0	x86	Linux	Xen	0.1	0.05	0.1	0.1	2
DC2	2	x86	Linux	Xen	0.1	0.05	0.1	0.1	1

Physical Hardware Details of Data Center : DC2

Id	Memory (Mb)	Storage (Mb)	Available BW	Number of Processors	Processor Speed	VM Policy
0	204800	100000000	1000000	4	10000	TIME_SHARED

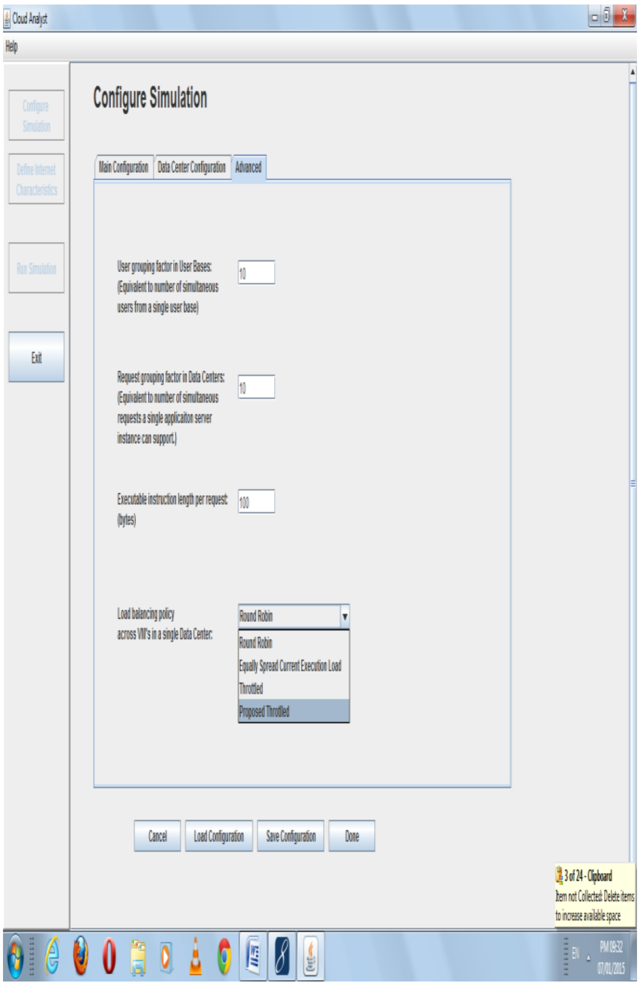
Figure 10

5.1.4 Internet Characteristics

In this component various internet characteristics are modeled simulation, which includes the amount of latency and bandwidth need to be assigned between regions, the amount of traffic, and current performance level information for the data centers.

5.1.5 VmLoadBalancer

The responsibility of this component is to allocate the load on various data centers according to the request generated by users. One of the f our given policies can be selected. The given policies are round robin algorithm, equally spread current execution load, throttled, proposed throttled.



Configure Simulation

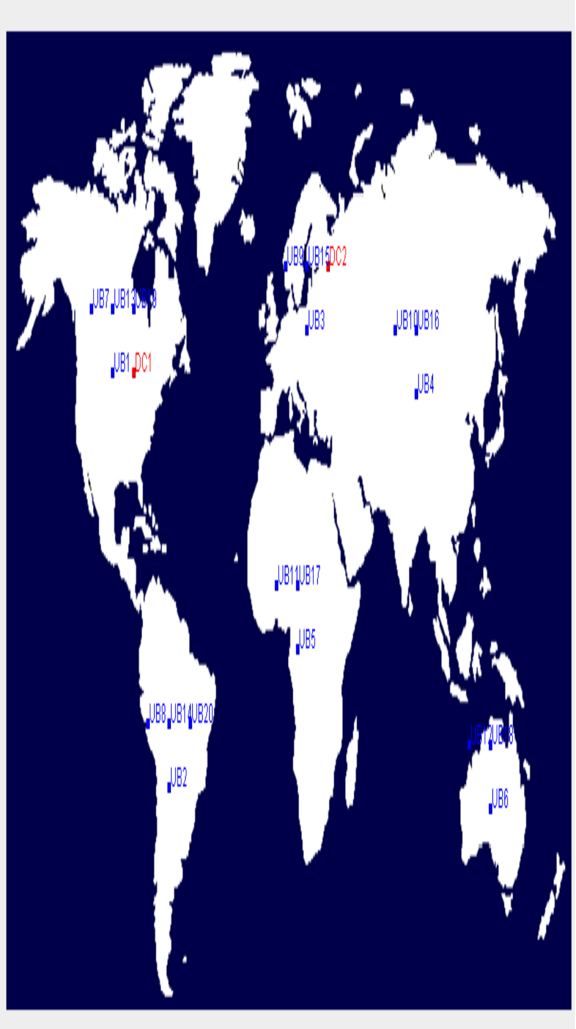
User grouping factor in User Bases: (Equivalent to number of simultaneous users from a single user base) 10

Request grouping factor in Data Centers: (Equivalent to number of simultaneous requests a single applicaiton server instance can support.) 10

Executable instruction length per request: (bytes) 100

Load balancing policy across VM's in a single Data Center: Round Robin

Figure 11Input Screen after Configurations

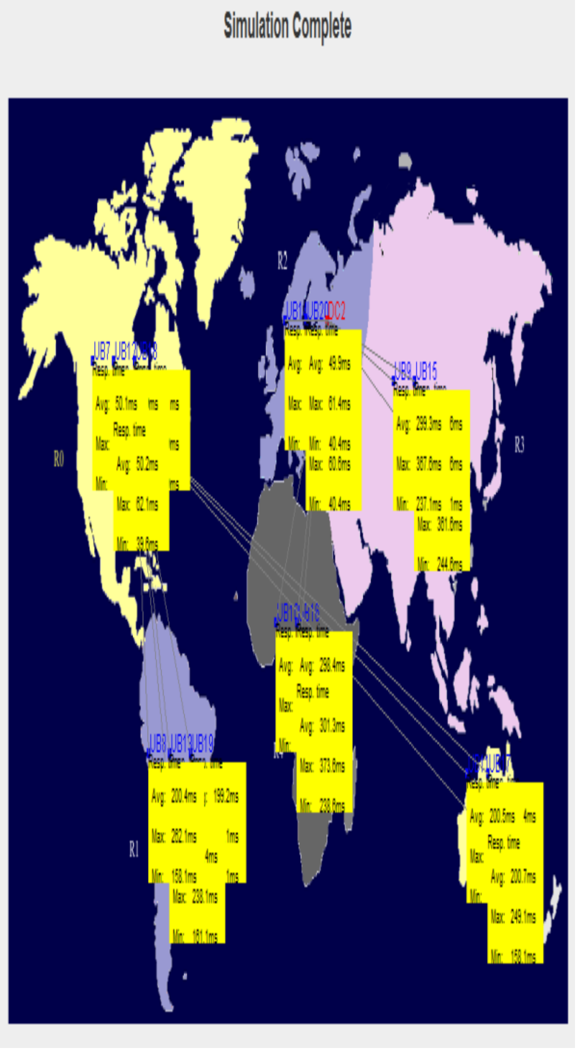


5.1.6 CloudAppServiceBroker

The responsibility of this component is to model the service brokers that handle traffic routing between user bases and data centers. The service broker can use one of the routing policies from the given three policies which are closest data center, optimize response time and reconfigure dynamically with load. The closest data center routes the traffic to the closest data center in terms of network latency from the source user base. The reconfigure dynamically with load routing policy works in the sense that whenever the performance of particular data center degrades below a given threshold value then the load of that data center is equally distributed among other data centers.

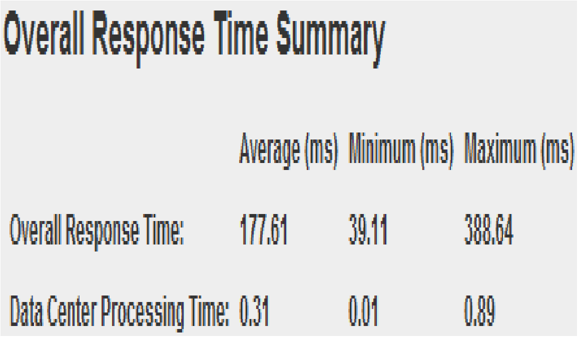
In order to analyze various load balancing policies configuration of the various component of the cloud analyst tool need to be done. We have set the parameters for the user base configuration, application deployment configuration, and data center configuration as shown in figure 6. As shown in figure the location of user bases has been defined in six different regions of the world. We have taken two data centers to handle the request of these users. On DC1 there are 15 VMs allocated, 5 VMs are allocated to DC2. Here we have taken 20 user bases.

Output Screen after Run Simulation



6. RESULTS AND RESPONSE TIME

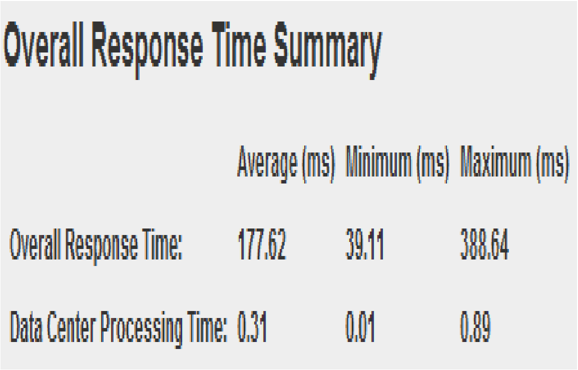
After performing the simulation the result computed by cloud analyst is as shown in the following figures. The above defined configuration has been used for each load balancing policy one by one and depending on that the result calculated for the metrics like response time, request processing time and cost in fulfilling the request has been shown. Parameters like average response time, data center service time and total cost of different data centers have taken for analysis.



Overall Response Time Summary

	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	177.61	39.11	388.64
Data Center Processing Time:	0.31	0.01	0.89

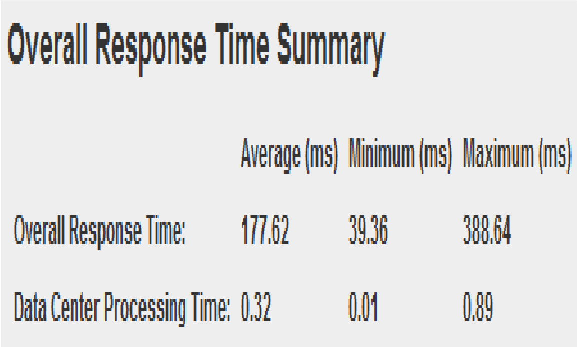
Figure 12: Response Time for RR with 20 User bases



Overall Response Time Summary

	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	177.62	39.11	388.64
Data Center Processing Time:	0.31	0.01	0.89

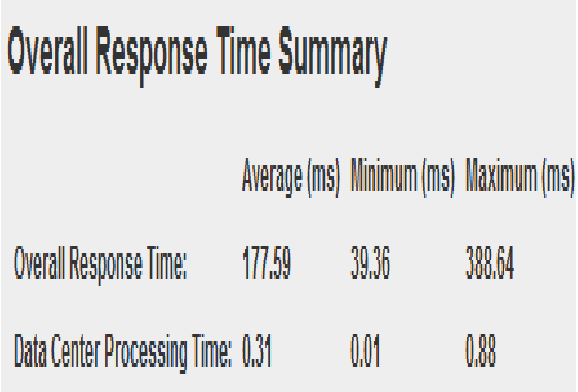
Figure 13: Response Time for ESCE with 20 User bases



Overall Response Time Summary

	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	177.62	39.36	388.64
Data Center Processing Time:	0.32	0.01	0.89

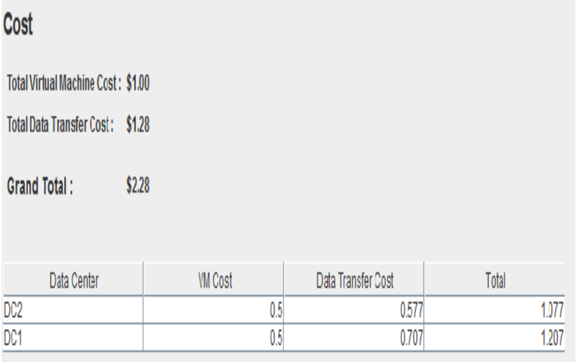
Figure 14: Response Time for TLB with 20 User bases



Overall Response Time Summary

	Average (ms)	Minimum (ms)	Maximum (ms)
Overall Response Time:	177.59	39.36	388.64
Data Center Processing Time:	0.31	0.01	0.88

Figure 15: Response Time for Proposed TLB with 20 User bases



Cost

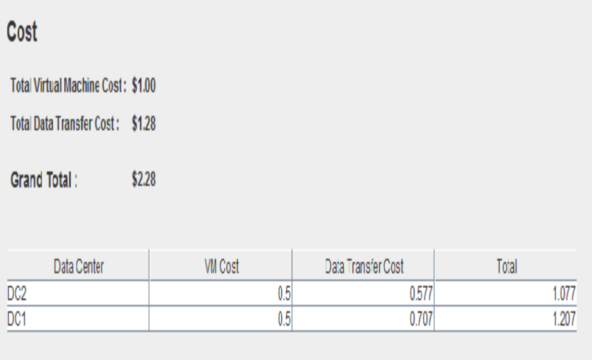
Total Virtual Machine Cost : $1.00

Total Data Transfer Cost : $1.28

Grand Total : $2.28

Data Center	VM Cost	Data Transfer Cost	Total
DC2	0.5	0.577	1.077
DC1	0.5	0.707	1.207

Figure 16: Processing Cost of RR



Cost

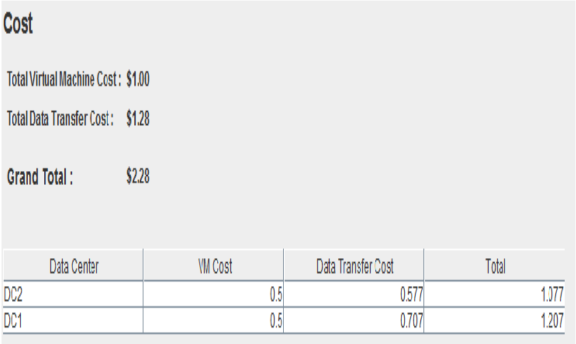
Total Virtual Machine Cost : $1.00

Total Data Transfer Cost : $1.28

Grand Total : $2.28

Data Center	VM Cost	Data Transfer Cost	Total
DC2	0.5	0.577	1.077
DC1	0.5	0.707	1.207

Figure 17: Processing Cost of ESCE



Cost

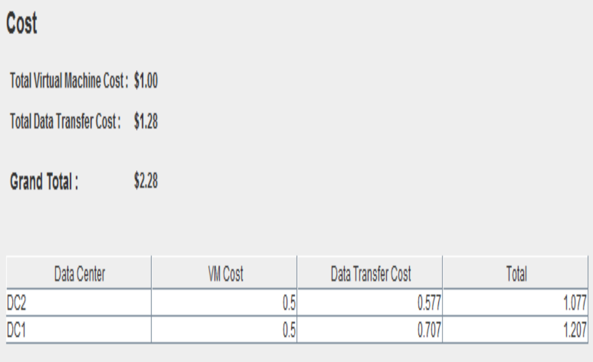
Total Virtual Machine Cost : $1.00

Total Data Transfer Cost : $1.28

Grand Total : $2.28

Data Center	VM Cost	Data Transfer Cost	Total
DC2	0.5	0.577	1.077
DC1	0.5	0.707	1.207

Figure 18: Processing Cost of TLB



Cost

Total Virtual Machine Cost : $1.00

Total Data Transfer Cost : $1.28

Grand Total : $2.28

Data Center	VM Cost	Data Transfer Cost	Total
DC2	0.5	0.577	1.077
DC1	0.5	0.707	1.207

Figure 19: Processing Cost of Efficient TLB

Experimental set up:
The above model & the proposed algorithm are implemented on Cloud Analyst. It is java based implementation tool. The parameters used in this experimental study are as follows:

1. Data Centers
2. UB (User Base)
3. VM (Virtual Machine)
4. Avg Data Center Processing Time
5. Min Data Center Processing Time
6. Max Data Center Processing Time
7. Avg Response Time
8. Min Response Time
9. Max Response Time
10. Total Cost

The comparative study is summarized in the following table:

The simulation Duration is 60 min & the service broker policy is Optimize Response Time.

Algorithms/ Parameters	**RR**	**ESCE**	**Throttled (TLB)**	**Efficient TLB**
Data Center	2	2	2	2
UB	20	20	20	20
VM	20	20	20	20
DC Proc. Time Avg (ms)	0.31	0.31	0.32	0.31
DC Proc. Time Min (ms)	0.01	0.01	0.01	0.01
DC Proc.Time Max (ms)	0.89	0.89	0.89	0.88
Overall response Time Avg (ms)	177.61	177.62	177.62	177.59
Overall response Time Min (ms)	39.11	39.11	39.36	39.36
Overall response Time Max (ms)	388.64	388.64	388.64	388.64
Overall Cost	2.28	2.28	2.28	2.28

From above table, it is clear that efficient throttled method is more efficient for the cloud load balancing.

7. CONCLUSION

As such cloud computing being wide area of research and one of the major topics of research is dynamic load balancing, so the following research will be focusing on algorithm consider mainly two parameters firstly, load on the server and secondly, current performance of server.

The goal of load balancing is to increase client satisfaction and maximize resource utilization and substantially increase the performance of the cloud system and minimizing the response time and reducing the number of job rejection.

Various new algorithms can be proposed for the load balancer so that the load is evenly distributed to every node resulting in better response time and user satisfaction.

We also conclude that Proposed Throttled VM load balancing algorithm is best among others.

REFERENCES

[1] G. Pallis, “Cloud Computing: The New Frontier of Internet Computing”, IEEE Journal of Internet Computing, Vol. 14, No. 5, September/October 2010, pages 70-73.

[2] Qi Zhang, Lu Cheng, Raouf Boutaba, “cloud computing: state of-the-art and research challenges”, 20th April 2010, Spinger, pp. 7-18.

[3] M. D. Dikaiakos, G. Pallis, D. Katsa, P. Mehra, and A. Vakali, “Cloud Computing: Distributed Internet Computing for IT and Scientific Research”, IEEE Journal of Internet Computing, Vol. 13, No. 5, September/October 2009, pages 10-13.

[4] Zenon Chaczko, Venkatesh Mahadevan, Shahrzad Aslanzadeh and Christopher Mcdermid,” Availability and Load Balancing in Cloud Computing” IPCSIT vol.14 (2011).

[5] Ram Prassd Pandhy (107CS046), P Goutam Prasad rao (107CS039). “Load balancing in cloud computing system” Department of computer science and engineering National Institute of Technology Rourkela, Rourkela-769008, Orissa, India May-2011.

[6] J. Sahoo, S. Mohapatra and R. lath “Virtualization: A survey on concepts, taxonomy and associated security issues” computer and network technology (ICCNT), IEEE, pp. 222-226. April 2010.

[7] Bhaskar. R, Deepu.S. R and Dr.B. S. Shylaja “Dynamic Allocation Method For Efficient Load Balancing In Virtual Machines For Cloud Computing Environment” September 2012.

[8] R.Shimonski. Windows 2000 & Windows server 2003 clustering and load balancing. Emeryville. McGraw-Hill Professional publishing, CA, USA (2003), p 2, 2003.

[9] R.X.T. and X. F.Z.A load balancing strategy based on the combination of static and dynamic, in database technology and applications (DBTA), 2010 2nd international workshops, (2010), pp. 1-4.

[10] Wenzheng Li, Hongyan Shi “Dynamic Load Balancing Algorithm Based on FCFS” IEEE, 2009. pp.1528-1531.

[11] Jiyni Li, Meikang Qui, Jain-Wei Niu, Yuchen, Zhong Ming “Adaptive resource allocation for preemptable jobs in cloud system”. IEEEInternational Conference on intelligent system design and applications, pp. 31-36, 2010.

[12] Sandeep Sharma, Sarabjit Singh, Meenakshi Sharma “Performance Analysis of Load Balancing Algorithms”, World Academy of
Science, Engineering and Technology, 38, 2008 pp. 269- 272.

[13] Bhathiya, Wickremasinghe.”Cloud Analyst: A Cloud Sim-based Visual Modeller for Analysing Cloud Computing Environments and Applications”, 2010, IEEE.

[14] http://www.cloudbus.org/cloudsim.