

Implementation and Result Analysis of Priority Based Load Balancing In Cloud Computing

Rahul Rathore, Bhumika Gupta, Vaibhav Sharma, Kamal Kumar Gola

Abstract—Cloud Computing is not a single terms, it is a combination of two word i.e. cloud and computing. Cloud in cloud computing is the set of hardware, software, network, storage, services, & interfaces that combine to deliver aspects of computing as a service. It has no significance with its name “Cloud”. It generally incorporates infrastructure as a service (IaaS), platform as a service (Pass), and software as a service (SaaS) which are delivered to end users over the Internet. To perform these entire thing load balancing is essential to improve the performance significantly, to provide a tendency of fault tolerance in case the system failures, to maintain the system constancy and to accommodate prospect variation in the system. Load balancing is the mechanism to distribute equal workload and a resource across all the nodes i.e. no single node is over loaded and others remains idle. It guarantees that each and every node has equal load. Now day’s product has been increased rapidly in the cloud computing environment, so we are proposed an algorithm which performs the load balancing on the basis of product priority.

Index Terms— Cloud computing, cloud services, cloud user, load balancing, load balancing policies.

1 INTRODUCTION

Cloud in cloud computing is the set of hardware, software, network, storage, services, & interfaces that combine to deliver aspects of computing as a service. Cloud computing provides different services such as IAAS, PAAS, SAAS for which users pay different amount as per their requirement. It allows people to do anything they want to do on a computer without the need for them to buy & build an IT infrastructure. To provide an efficient use of these resources and services to the end users “Computing” is done. Computing is done on the basis of service level agreement (SLA). These services are available to the user’s On-Demand basis in pay per use manner

.Load balancing is the major area of concern in cloud computing. It is a mechanism that distributes the workload evenly across all the nodes in the whole cloud to avoid a situation where some nodes are heavily loaded while others are idle or doing little work. It helps to achieve a high user satisfaction and resource utilization ratio, hence improving the overall performance and resource utility of the system. Load balancing is the process of reassigning the total loads to the individual nodes of the collective system to make the best response time and also good utilization of the resources. Cloud computing is a service providing facility over the internet in which the load balancing is the one of the challenging task. Various methods are to be used to make a better system by allocating the loads to the nodes in a balancing manner but due to network congestion, bandwidth usage etc, there were problems are occurred. These problems were solved by some of the existing techniques. A load balancing algorithm which is dynamic in nature does not require

any previous information behaviour of the system, that is, it depends on the current behaviour of the system. There are various goals that related to the load balancing such as to improve the performance substantially, to maintain the system stability etc. Depending on the current state of the system, load balancing algorithms can be categorized into two types they are static and dynamic algorithms. In static environment advanced provisioning is done in which the providers prepare the appropriate resources in advance of the start of services according to the contracts between user and service provider



Figure 1.1 Cloud Computing

2 DEPLOYMENTS MODELS OF CLOUD

2.1 PRIVATE CLOUD The cloud infrastructure is operated exclusively for an organization. In this model the organization keep their critical information and services in their control. It deployments made inside the organization firewall’s. It may be managed by the organization or a third party. Private cloud offers some of

the advantage of public cloud computing environment, such scalability of resources.

2.2 PUBLIC CLOUD The term public does not mean i.e. it is free, even though it can be free or fairly inexpensive to use and that a user data is publically visible i.e. public cloud vendors typically provide an access control mechanism for their users. The cloud infrastructure is made available to the general public or a large industry group and is owned by an organization selling cloud services.

2.3 COMMUNITY CLOUD The cloud infrastructure is shared by several organizations and supports a specific community that has common objective. It may be handled by the organizations or other party to smoothly access the services of cloud.

2.4 HYBRID CLOUD The cloud infrastructure is a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability. In this model client typically outsource non business critical information & processing to public cloud, while having control on critical information and data.

3 SERVICES PROVIDED BY CLOUDS

Cloud provides services in three different ways.

Software as a Service (SaaS)

Platform as a Service (PaaS)

Infrastructure as a Service (IaaS)

3.1 SOFTWARE AS A SERVICE (SAAS)

It is the ability to offer services over the internet to the end user. These services can be used from different devices which can run upon different platforms. The user does not require to manage cloud infrastructure, including storage, servers, network operating systems.

3.2 PLATFORM AS A SERVICE (PAAS)

It is the ability to offer services over the internet to the end user i.e the intermediary stage is to install on the cloud Infrastructure, the user develop or access applications developed using programming languages and tools supported by the service provider. The user does not require to manage the underlying cloud infrastructure, including storage, servers, network operating systems.

3.3 INFRASTRUCTURE AS A SERVICE (IAAS)

It is the ability to provisioning and re-provisioning the resources such as storage, networks, operating system to the end user as per the requirement. Where the user is

capable to install and execute different software, which includes operating systems and applications.

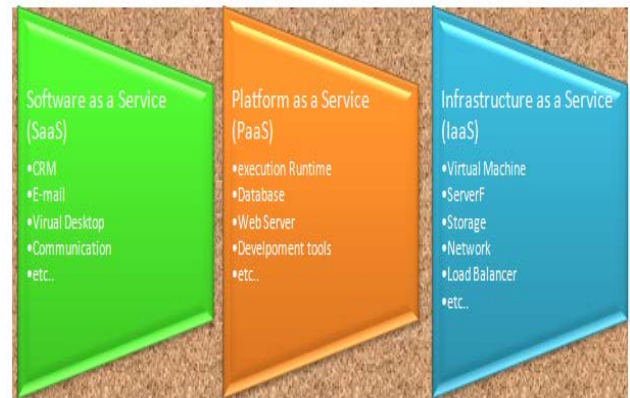


Figure 3.1 Services of Cloud

4 LOAD BALANCING

Load balancing is the mechanism of distributing the total load to the individual nodes of the system in such a manner that each & every node effectively utilize the resources and minimize the response time. It also solves the problem in which some of the nodes are over loaded or some of the nodes are under loaded or remains idle. A load balancing algorithm which is dynamic in nature does not consider the previous state or behaviour of the system, that is, it depends on the present behaviour of the system. The key attributes used in developing such algorithm are : performance of system, comparison of load, stability of different system, interaction between the nodes, estimation of load, selecting of nodes, nature of work to be transferred, and many other ones [4] . This load considered can be in terms of CPU load, amount of memory used, delay or Network load

4.1 GOALS OF LOAD BALANCING

- To improve the performance significantly
- To improve the response time.
- To provide a tendency of fault tolerance in case of the system failures To maintain the system reliability
- To accommodate unpredicted variation of the system

4.2 POLICIES OR STRATEGIES IN DYNAMIC LOAD BALANCING

- Transfer Policy: It selects a job for transferring from a local node to a remote node is referred to as Transfer policy or Transfer strategy.

- Selection Policy: It identifies the processor which is best fit for processing the request or job for load exchange.
- Location Policy: It selects a destination node for a transferred task is referred to as location policy or Location strategy.
- Information Policy: It is responsible for gathering information about the nodes in the system is referred to as Information policy or Information strategy.

4.3 LOAD BALANCING ON THE BASIS OF CLOUD ENVIRONMENT

Cloud computing can have either static or dynamic environment based upon how developer configures the cloud demanded by the cloud provider.

STATIC ENVIRONMENT

In static environment the cloud provider installs homogeneous resources. In static environment advanced provisioning is done in which the providers prepare the appropriate resources in advance of the start of services according to the contracts between user and service provider. The resources in the cloud are reserved and not scalable because of the static environment. In this scenario, the cloud requires prior knowledge of nodes capacity, processing power, memory, performance and statistics of user requirements. These user requirements are not variable at run-time. Algorithms proposed to achieve load balancing in static environment cannot adjust to the run time changes in load.

DYNAMIC ENVIRONMENT

In dynamic environment the cloud provider installs heterogeneous resources. In dynamic environment dynamic provisioning is done in which the service provider allocates more resources as they are needed to the user & removed them when they are not used. The resources are scalable in dynamic environment. In this environment cloud does not need any former knowledge. The requirements of the users are scalable i.e. they may change at run-time. Algorithm proposed to achieve load balancing in dynamic environment can easily adjust to run time changes in load. Dynamic environment is difficult to be simulated but is highly adjustable with cloud computing environment.

4.4 LOAD BALANCING BASED ON SPATIAL DISTRIBUTION OF NODES

Nodes in the cloud are distributed in discrete manner. Hence the node that makes the provisioning decision also administrate the category of algorithm to be used. There

are three types of algorithms that specify which node is responsible for balancing of load in cloud computing environment.

CENTRALIZED LOAD BALANCING

In centralized load balancing technique all the resource provisioning and scheduling decision are made by a centralized node. This node is responsible for keeping record of entire cloud network knowledge and can apply static or dynamic approach for load balancing. This technique minimize the time required to analyze different cloud resources but increase a great overhead on the centralized node. Centralized node reduces the tendency of fault tolerance, in this scenario as failure intensity of the overhead centralized node is high and recovery might not be easy in case of centralized node failure.

DISTRIBUTED LOAD BALANCING

In distributed load balancing technique, no specific node is responsible for making decision and resource provisioning. Monitoring of network is done by multiple nodes to make accurate and efficient load balancing. Every node in the network maintains local knowledge base to ensure efficient allotment of tasks in static environment and re-allotment in dynamic environment. In this scenario, the system is fault tolerant and balanced because no single node is overloaded to make load balancing decision.

HIERARCHICAL LOAD BALANCING

Hierarchical load balancing used the concept of parent child relationship re involves different levels of the cloud in load balancing decision. These can be done using the model of tree data structure wherein every node in the tree is balanced under the supervision of its parent node. Provisioning or scheduling decision is based upon the information gathered by the parent node.

5 LITERATURE REVIEW

In this section, the significant involvements of load balancing in cloud computing, as stated in the literature are discussed. A. Khiyaita et al. [1], his approach gave an overview of load balancing in cloud computing, types of load balancing algorithms depends on the system capabilities and configuration such as network topologies. While in [2] authors talked about the existing techniques whose goals was to diminish the associated overhead, to increase response time of the service and improving performance of the technique. The authors also gave details about different parameters, used to compare the presented techniques.

D. A. Menasce et al. [3] told about the concept of cloud computing, its pro's and con's and described various

presented cloud computing platforms. He also stated the results of quantitative experiments performed on PlanetLab. Kuo-Qin Yan et al. [4] his approach proposed a scheduling algorithm to maintain the load balancing in the cloud computing environment, his algorithm was a combination of the capabilities of both OLB (Opportunistic Load Balancing) [5] and LBMM (Load Balance Min-Min) [6] scheduling algorithms, and provide more efficient way of scheduling.

In [7], the aim is to get the best cloud resource while considering Co-operative Power attentive Scheduled Load Balancing, key to the Cloud load balancing challenges.

The algorithm stated in this paper, utilizes the advantages of both distributed and centralized techniques of computing. Such as efficiency of centralized technique and the energy efficiency and capability of fault-tolerance in distributed approach is effectively used in this algorithm.

In PALB [8] this approach, the author estimated the utilization percentages of each node. This helps in determining the nodes which are in used or remains idle. The algorithm divided into three parts: Balancing section, Upscale section and Downscale section. Balancing section is answerable for determining where virtual machines will be instantiated depends on utilization percentages. The upscale section initiates the other computing nodes. And the downscale section power off the unused compute nodes. This algorithm is guaranteed to reduce the overall consumption of power while maintaining the accessibility of resources in comparison to other load balancing algorithms.

Raul Alonso-Calvo et al. [9] discussed about the managing of large gathering of images in companies and institutions. A cloud computing service and its application for gathering and analysis of large images have been generated and the data operations are modified for working in a distributed manner by using various sub-images that can be stored and processed independently by different agents in the system, facilitating processing very-large images in a parallel manner. This work can be viewed as a different way of load balancing in cloud computing. Excluding from accessibility of resources, other factors like scalability of resources and power consumption was also important issues in load balancing that cannot be ignored. Load balancing techniques should be such as to obtain measurable improvements in resource utilization and availability of resources in the cloud computing environment [10]. Alexandru Iosup et al. [11] examined the performance of cloud computing services for scientific computing workloads and compute the presence in real scientific computing workloads of Many-Task Computing (MTC) users, i.e., of users who

make use of loosely coupled applications consist of many tasks to attain their scientific goals. They also perform a practical evaluation of the performance of four marketable cloud computing services. Srinivas Seth et al. [12] proposed a of fuzzy logic in load balancing the cloud computing environment. This algorithm works on two attributes processor speed and assigned load to virtual machine, to balance the overhead using fuzzy logic, while in [13], this authors proposed a new fuzzy logic based dynamic load balancing algorithm with additional attributes, bandwidth usage, - memory usage, disk space usage and virtual machine status and named it as Fuzzy Active Monitoring Load Balancer (FAMLB).

Milan E. Sokile [14], told about various load balancing approach in a distributed environment, named as diffusive load, static or fixed, round robin approach and shortest queue in various users environments. Experimental analysis have been done displaying diffusive load balancing is more efficient than static and round robin load balancing technique in a dynamic environment.

Ankush P. Deshmukh and Prof. Kumarswamy Pamu [15], discussed on the diversity of load balancing policy, algorithms and methods. The research also demonstrates that the dynamic load balancing is far better than other static load balancing techniques. Efficient load balancing can clearly give major performance benefits [16]. A Network Processor built of a number of on-chip processors to take out packet level parallel processing operations, guarantee good load balancing among the processors. This process enhanced the throughput of the system. In this paper, the authors first propose an Ordered Round Robin (ORR) scheme to schedule packets in a heterogeneous network processor, assuming that the processed loads from the processors are perfectly ordered. This paper examines the throughput and get expressions for the batch size, scheduling time and maximum number of schedulable processors.

Jaspreet Kaur [17] has proposed a technique of active virtual machine load balancer algorithm to determine a appropriate virtual machine in less amount of time. She performed simulation to demonstrate the comparative analysis of round robin and equal spread present execution policies of load balancing with different service broker policies for data center in a cloud computing environment.

Zhang Bo et al. [18], proposed an algorithm which provide additional capacity to the dynamic load balance mechanism in the cloud computing environment. The experiments express that the algorithm has gained a better load balancing amount and used less amount of time in loading all request.

Soumya Ray and Ajanta De Sarkar [19] have discussed about various load balancing algorithm, their analysis is

based on MIPS vs. VM and MIPS vs. HOST. Their results show that these algorithms can definitely enhance the response time in order of magnitude, with respect to number of VMs in Datacenter. Variation in MIPS will affect the response time. Increasing MIPS vs. VM reduce the response time. In order to handle the arbitrary selection based load distribution problem, dynamic load balancing algorithm can be used as the future work to calculate various attributes for balancing. In [20], the authors proposed an algorithm to distribute work load among various nodes in cloud environment, by the use of Ant Colony Optimization (ACO). It uses the concept of genetic algorithm to equally distribute the work load among various nodes.

Shridhar G. Domanal and G. Ram Mohana Reddy [21] have proposed a local optimized load balancing approach for distributing incoming job request uniformly between the servers or virtual machines. They analyzed the performance of their algorithm using the Cloud Analyst simulator. Further, they also compared their approach with Round Robin [22] and Throttled algorithm [23]. A similar work was done by Shridhar G. Domanal and G. Ram Mohana [21] using CloudSim and Virtual Cloud simulators. In [24], the authors have analyzed various policies in combination with different load balancing algorithms using a tool called Cloud Analyst. They presented various variants of Round Robin load balancing algorithm, demonstrating the pros and cons of each. The Dynamic Round Robin algorithm is an improvement over static Round Robin algorithm [25], this paper analyzed the Dynamic Round Robin approach with different attributes like host bandwidth, VM image size and, cloudlet length. Results have been carried out by using CloudSim simulator. In [26], the authors Ching-Chi Lin, Pangfeng Liu, Jan-Jan Wu, have proposed a new Dynamic Round Robin (DRR) algorithm for energy-aware virtual machine scheduling and consolidation. This algorithm is judge against with current existing algorithms, like Greedy, Round Robin approach and Power saving strategies used in Eucalyptus.

6. PROPOSED ALGORITHM

6.1 USER REGISTRATION If the user is already registered, then they have the following attributes.

- i) User_id(UID)
- ii) Number of virtual machines required(NVM)
- iii) Execution time of the request(ET)
- iv) Customer registration year(CRY)
- v) Arrival time of request(ATR)
- vi) Cost of the product(CP)

If the user is new then it will registered itself with the service provider, the service provider provides the two attributes to the new user i.e. user_id and CRY

6.2 GENERATION OF PRIORITY LIST

- i) The priority is assigned to the request is based on the above listed attributes of the user.
- ii) The priority is based on the weightage of each attribute; weightage may vary according to the service provider. The Weightage of attribute may be defined as

Table 6.2.1 Weightage Distribution of CRY

Attribute CRY	Total Weightage 20%
0-1	5%
2-3	10%
3-4	15%
Greater than or equals to 5	20%

Table 6.2.2 Weightage Distribution of CP

Attribute CP	Total Weightage 30%
Cost <= 25000	5%
25001>=Cost<= 50000	10%
50001>=Cost<=100000	15%
100001>=Cost<= 150000	20%
150001>=Cost <= 200000	25%
Cost >= 200001	30%

Table 6.2.3. Weightage Distribution of NVM

Attribute NVM	Total Weightage 25%
1-2	25%
3-4	20%
5-6	15%
7-8	10%
9-10 or greater than 10	5%

Table 6.2.4. Weightage Distribution of ET

Attribute ET	Total Weightage 25%
1-10 minutes	25%
11-15 minutes	20%
16-20 minutes	15%
21-25 minutes	10%
26-30 minutes or greater than 30	5%

- iii) If there are 10 requests from different users then each request have their UID, NVM, ET, CRY, CP and ATR.
- iv) If two requests having same Weightage then the priority will decided on the basis of their arrival time (AT) i.e. the request which comes first will be served first.

Let R1, R2, R3 and R4 are the requests from different users then:

R1 having:

Table 6.2.5. Value calculation for request R1.

Attribute	CR Y	CP	ET	NV M	AT R	Total
Weightage	1	50,000	15mins	5	9:05 am	50
Value	5	10	20	15	-	

R2 having:

Table 6.2.6. Value calculation for request R2.

Attribute	CR Y	CP	ET	NV M	AT R	Total
Weightage	3	1,50,000	10mins	8	10:00 am	65
Value	10	20	25	10	-	

R3 having:

Table 6.2.7. Value calculation for request R3.

Attribute	CR Y	CP	ET	NV M	AT R	Total
Weightage	5	2,00,000	20mins	7	10:35 am	70
Value	20	25	15	10	-	

R4 having:

Table 6.2.8. Value calculation for request R4.

Attribute	CR Y	CP	ET	NV M	AT R	Total
Weightage	3	1,20,000	10mins	7	8:48 am	65
Value	10	20	25	10	-	

R3 having the highest value so, its having the higher priority over R1, R2 and R4, here R2 and R4 having the equal value so their priority will be decided by the attribute ATR, ATR of R4 is less than R2 therefore R4 having higher priority over R2; R1 having the lowest value so its priority is low.

The generated priority list is:

Table 6.2.9. Priority list of requests

Priority of Request	Value
R3	70
R4	65
R2	65
R1	50

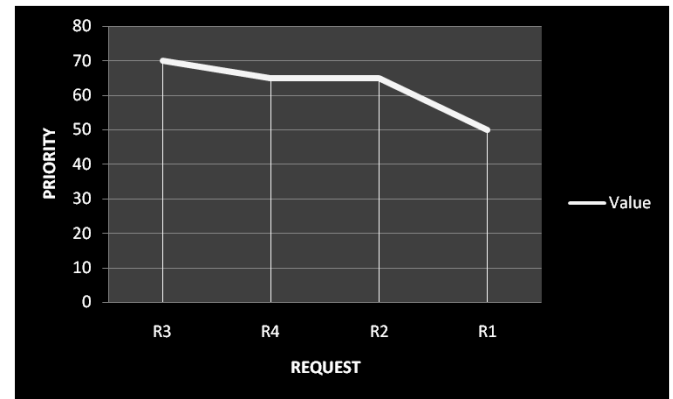


Figure 6.2.2. Priority Assignment to Requests

6.3 SERVER LOAD CALCULATION

- Each data centre has numbers of queues which are equivalent to the number of virtual machines associated with the data centre.
- Each queue associated with each virtual machine. These queues keep the record of virtual machine status i.e. how many requests a virtual machine have.
- These queues help data centre to make decision of transferring the request to the less loaded virtual machine.
- If the queues at data centre are full then the new request will be queued up in a waiting list.
- When any VM's completes its execution, then the associated queue update its status and informed the data centre about that particular VM's is free to execute new request.

6.4 Server Allocation Process

- Read the number of nodes required by the incoming request.
- Find the server to which if the job is assigned for execution will leave the least number of idle nodes.
- If all servers are busy queue the job in the remote wait list.
- If not assign the job to the selected server.

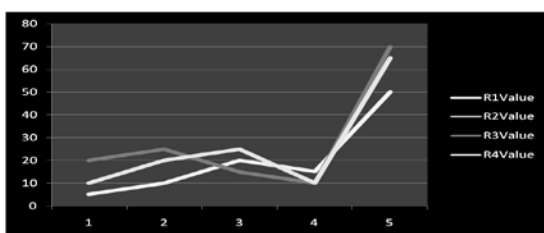


Figure 6.2.1 Value Calculation of Request

- v. If a few nodes of a server are freed then take a job out of the wait list and jump to step ii.
- vi. Stop if there are no more queued jobs and no jobs being executed.

6.5 Waiting List

- i. The request which is not allotted to any data centre is queued into a wait list sorted in decreasing order according to their priority value.
- ii. When a request completes its execution, the data centre deallocates the VM's to execute another request.
- iii. Now the algorithm checks for request in the waiting list. If the node requirement matches then that request is taken out of the wait list and a VM's is allocated to it.

7. IMPLEMENTATION

In this implementation we use different data centers named as KKG, BG, Rahul, RR1 located in different geographical regions. Each data center provides service to requests made under its region. If the queue of data center is full then the request is forwarded to the centrally located data center i.e. geographically at the centroid of all the data centers. Following are the test cases and results of the simulation.

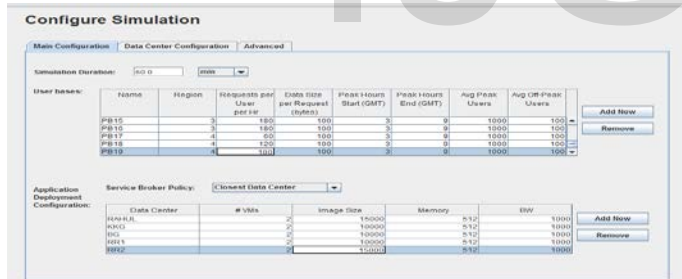


Figure 7.1 Simulating Request from users

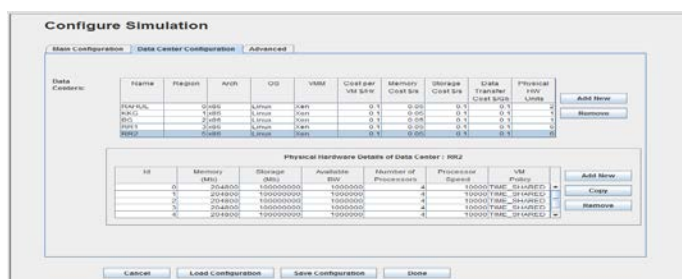


Figure 7.2 Simulating Request from users

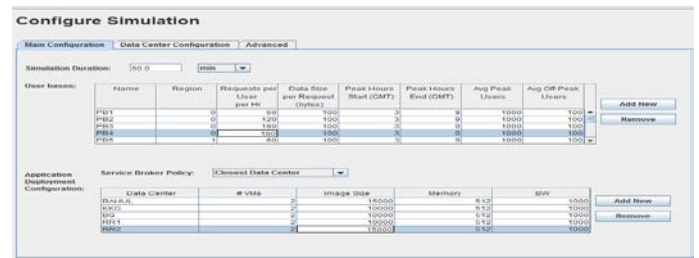


Figure 7.3 Simulating Request from users

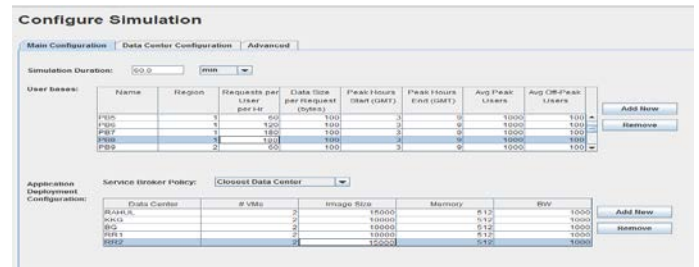


Figure 7.4 Simulating Request from users

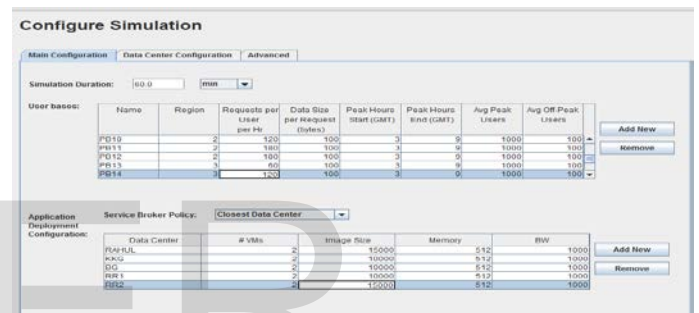


Figure 7.5 Simulating Request from users

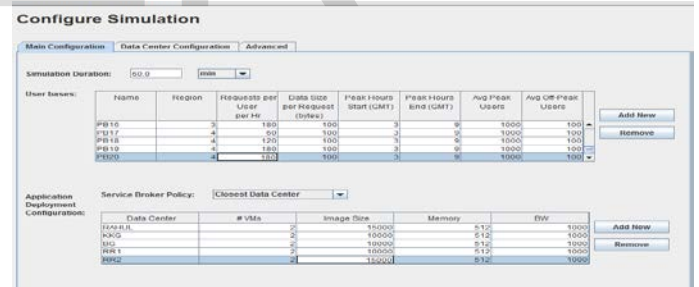


Figure 7.6 Simulating Request from users



Figure 7.7 Data Centre an virtual Machines located at Geographical Regions



Figure 7.8 Request Processing at Data Centre

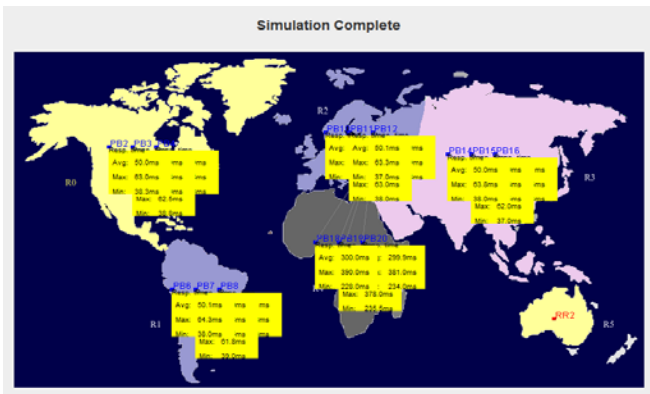


Figure 7.9 Responses From the Data Centres

Userbase	Avg (ms)	Min (ms)	Max (ms)
PB10	50.06	36.53	62.28
PB11	50.08	37.02	63.28
PB12	49.98	34.78	62.03
PB13	50.06	37.03	62.03
PB14	49.99	38.03	63.78
PB15	50.03	38.53	65.78
PB16	50.05	36.28	63.53
PB17	299.84	235.53	378.04
PB18	300.00	228.03	390.03
PB19	300.10	225.03	378.03
PB1	50.02	38.78	62.53
PB20	299.85	234.03	381.04
PB2	49.95	38.27	63.03
PB3	49.98	38.03	65.77
PB4	50.02	37.77	64.78
PB5	50.17	39.04	61.80
PB6	50.13	38.03	64.29
PB7	50.02	37.54	64.29
PB8	50.08	36.79	65.79
PB9	50.00	38.02	63.03

Hourly Response Times of individual user base.

PB1

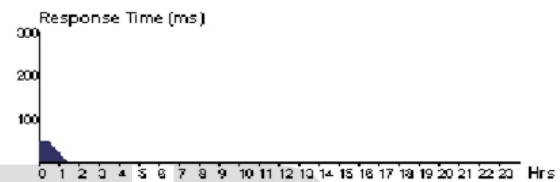


Figure 8.1 Hourly Response Time User Base PB1

PB10

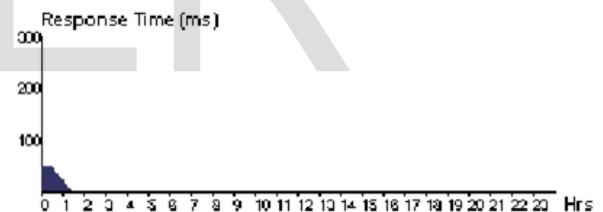


Figure 8.2 Hourly Response Time User Base PB10

PB11

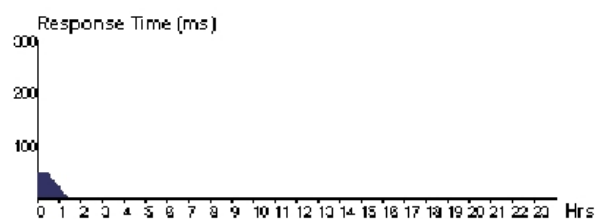


Figure 8.3 Hourly Response Time User Base PB11

PB12

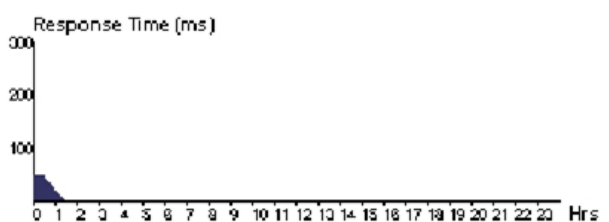


Figure 8.4 Hourly Response Time User Base PB12

8. RESULTS

The results show the response & processing time taken by the data center to each request from different regions, each request have priority and it is processed according to their priority . In this work 5 data centers are used, which are located in different regions.

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	90.89	34.78	390.03
Data Center processing time:	0.38	0.00	0.93

Response Time by Region to user base

Table 8.1 Hourly Response time

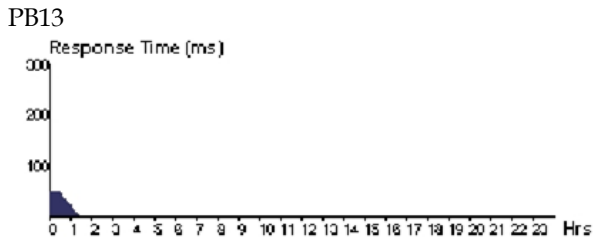


Figure 8.5 Hourly Response Time User Base PB13

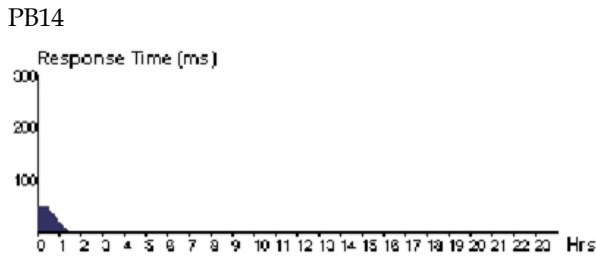


Figure 8.6 Hourly Response Time User Base PB14

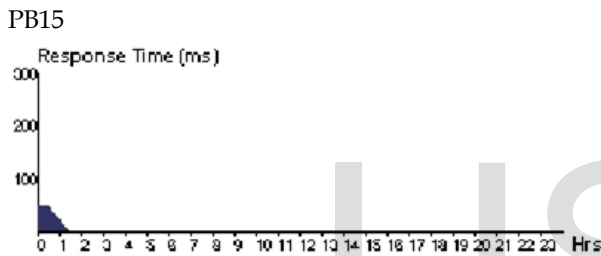


Figure 8.7 Hourly Response Time User Base PB15

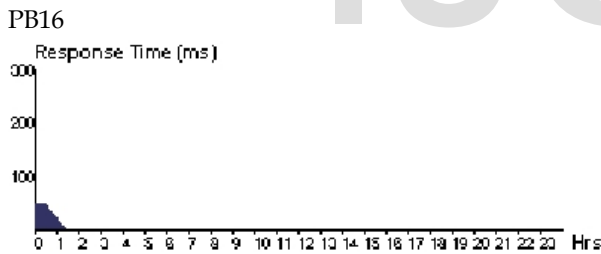


Figure 8.8 Hourly Response Time User Base PB16

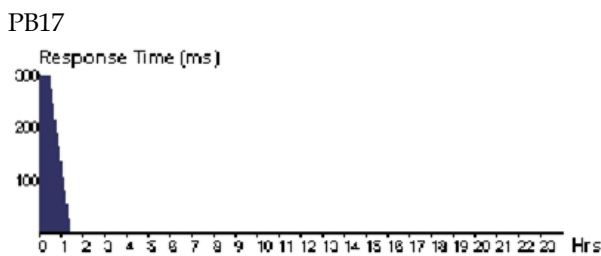


Figure 8.9 Hourly Response Time User Base PB17

PB18

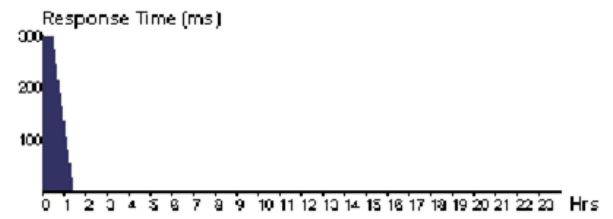


Figure 8.10 Hourly Response Time User Base PB18

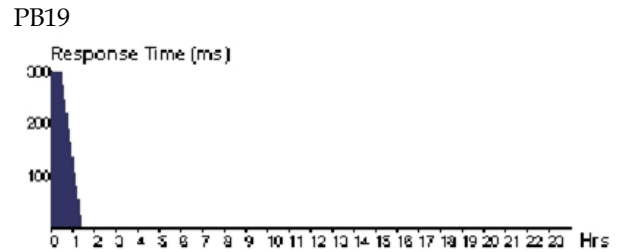


Figure 8.11 Hourly Response Time User Base PB19

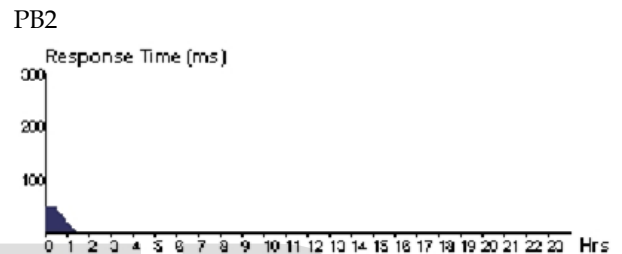


Figure 8.12 Hourly Response Time User Base PB2

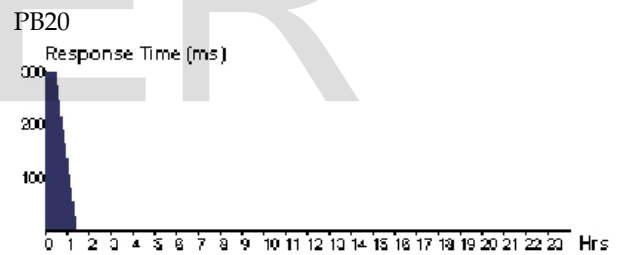


Figure 8.13 Hourly Response Time User Base PB20

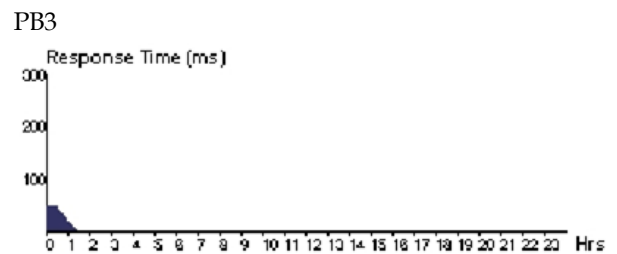


Figure 8.14 Hourly Response Time User Base PB3

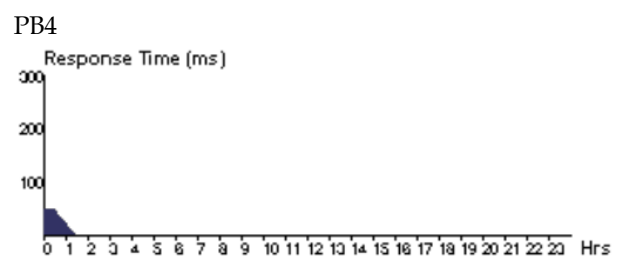


Figure 8.15 Hourly Response Time User Base PB4

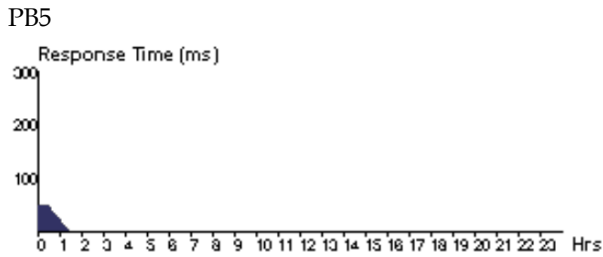


Figure 8.16 Hourly Response Time User Base PB5

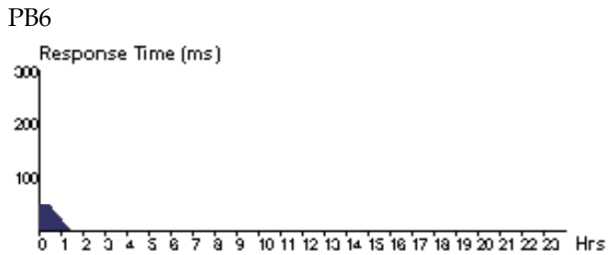


Figure 8.17 Hourly Response Time User Base PB6

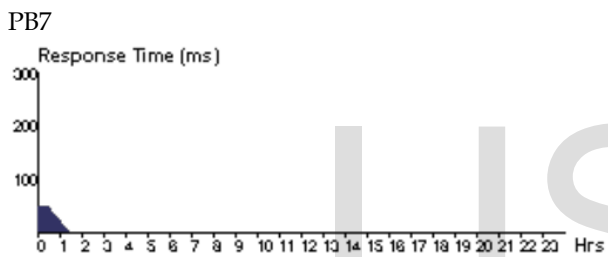


Figure 8.18 Hourly Response Time User Base PB7

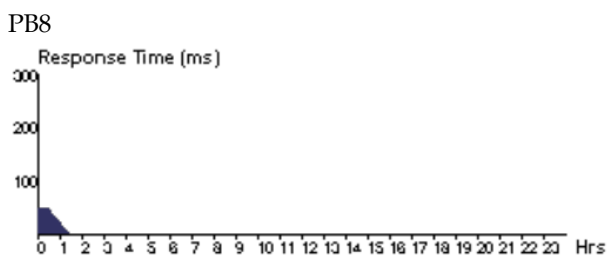


Figure 8.19 Hourly Response Time User Base PB8

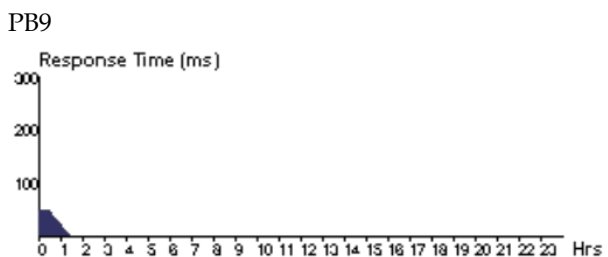


Figure 8.20 Hourly Response Time User Base PB9

8.1 DATA CENTER REQUEST SERVICING TIMES

Table 8.1.1 Datacenter Response Time

Data Center	Avg (ms)	Min (ms)	Max (ms)
BG	0.35	0.00	0.86
KKG	0.41	0.01	0.87
RAHUL	0.41	0.00	0.89
RR1	0.40	0.01	0.93
RR2	0.00	0.00	0.00

Data Center Hourly Loading

BG

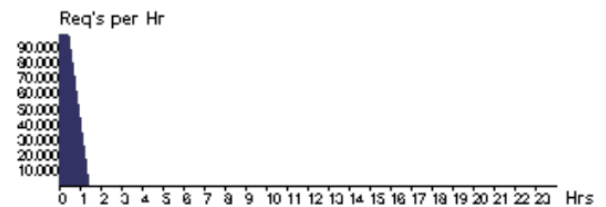


Figure 8.1.1 Hourly Loading Data Center BG

KKG

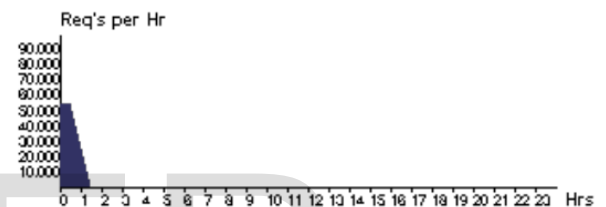


Figure 8.1.2 Hourly Loading Data Center KKG

RAHUL

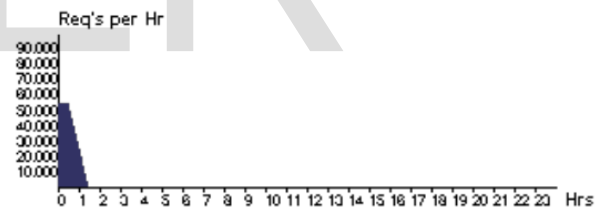


Figure 8.1.3 Hourly Loading Data Center RAHUL

RR1

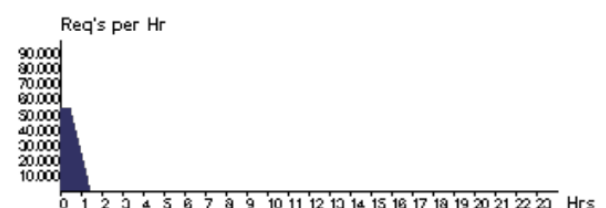


Figure 8.1.4 Hourly Loading Data Center RR1

9. CONCLUSION

By using this algorithm the service provider will ensure that the privileged users will be served first. From the user's point of view, user will get access to resource on priority basis, so if a user wants to get the access of resources he/she has to attain higher priority. To attain higher priority the user has to take care of the parameters

on which priority is calculated. This algorithm ensures that the user with higher priority will be served first in comparison to others user. To enhance the productivity of the service provider, it is necessary that its delivered services to their customer on time with maximum throughput. The simulation demonstrate the the higher priority user gets the services first. Today's scenario it is mandatory for service provider to have good relationship with their privileged user, this algorithm provide the services to the privileged user to make this relationship between user and service provider in cloud computing environment.

10. REFERENCES

- [1] A. Khiyaita, M. Zbakh, H. El Bakkali, and D. El Kettani, "Load balancing cloud computing: state of art," in *Network Security and Systems (JNS2)*, 2012 National Days of, pp. 106-109, IEEE, 2012.
- [2] N. J. Kansal and I.Chana, "Existing load balancing techniques in cloud computing: A systematic review," *Journal of Information Systems & Communication*, vol. 3, no. 1, 2012.
- [3] D. A. Menasc'e and P. Ngo, "Understanding cloud computing: Experimentation and capacity planning," in *Computer Measurement Group Conference*, 2009.
- [4] S.-C. Wang, K.-Q. Yan, W.-P. Liao, and S.-S. Wang, "Towards a load balancing in a three-level cloud computing network," in *Computer Science and Information Technology (ICCSIT)*, 2010 3rd IEEE International Conference on, vol. 1, pp. 108-113, IEEE, 2010.
- [5] T. D. Braun, H. J. Siegel, N. Beck, L. L. B'ol'oni, M. Maheswaran, A. I. Reuther, J. P. Robertson, M. D. Theys, B. Yao, D. Hensgen, et al., "A comparison of eleven static heuristics for mapping a class of independent tasks onto heterogeneous distributed computing systems," *Journal of Parallel and Distributed computing*, vol. 61, no. 6, pp. 810-837, 2001.
- [6] T. Kokilavani and D. Amalarethnam, "Load balanced min-min algorithm for static meta-task scheduling in grid computing," *International Journal of Computer Applications*, vol. 20, no. 2, 2011.
- [7] T. Anandharajan and M. Bhagyaveni, "Co-operative scheduled energy aware load-balancing technique for an efficient computational cloud," *International Journal of Computer Science Issues (IJCSI)*, vol. 8, no. 2, 2011.
- [8] J. M. Galloway, K. L. Smith, and S. S. Vrbsky, "Power aware load balancing for cloud computing," in *Proceedings of the World Congress on Engineering and Computer Science*, vol. 1, pp. 19-21, 2011.
- [9] R. Alonso-Calvo, J. Crespo, M. Garcia-Remesal, A. Anguita, and V. Maojo, "On distributing load in cloud computing: A real application for very-large image datasets," *Procedia Computer Science*, vol. 1, no. 1, pp. 2669-2677, 2010.
- [10] Z. Chaczko, V. Mahadevan, S. Aslanzadeh, and C. Mcdermid, "Availability and load balancing in cloud computing," in *International Conference on Computer and Software Modeling*, Singapore, 24 Comparative Study on Load Balancing Techniques in Cloud Computing vol. 14, 2011.
- [11] A. Iosup, S. Ostermann, M. N. Yigitbasi, R. Prodan, T. Fahringer, and D. H. Epema, "Performance analysis of cloud computing services for many-tasks scientific computing," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 22, no. 6, pp. 931-945, 2011.
- [12] S. Sethi, A. Sahu, and S. K. Jena, "Efficient load balancing in cloud computing using fuzzy logic," *IOSR Journal of Engineering*, vol. 2, no. 7, pp. 65-71, 2012.
- [13] Z. Nine, M. SQ, M. Azad, A. Kalam, S. Abdullah, and R. M. Rahman, "Fuzzy logic based dynamic load balancing in virtualized data centers," in *Fuzzy Systems (FUZZ)*, 2013 IEEE International Conference on, pp. 1-7, IEEE, 2013.
- [14] M. E. Soklic, "Simulation of load balancing algorithms: a comparative study," *ACM SIGCSE Bulletin*, vol. 34, no. 4, pp. 138-141, 2002.
- [15] A. P. Deshmukh and K. Pamu, "Applying load balancing: A dynamic approach," *International Journal*, vol. 2, no. 6, 2012.
- [16] J. Yao, J. Guo, and L. N. Bhuyan, "Ordered round-robin: An efficient sequence preserving packet scheduler," *Computers, IEEE Transactions on*, vol. 57, no. 12, pp. 1690-1703, 2008.
- [17] J. Kaur, "Comparison of load balancing algorithms in a cloud," *International Journal of Engineering Research and Applications*, vol. 2, no. 3, pp. 1169-173, 2012.
- [18] Z. Bo, G. Ji, and A. Jieqing, "Cloud loading balance algorithm," in *Information Science and Engineering (ICISE)*, 2010 2nd International Conference on, pp. 5001-5004, IEEE, 2010.
- [19] S. Ray and A. De Sarkar, "Execution analysis of load balancing algorithms in cloud computing environment," *International Journal on Cloud Computing: Services & Architecture*, vol. 2, no. 5, 2012.
- [20] K. Nishant, P. Sharma, V. Krishna, C. Gupta, K. P. Singh, N. Nitin, and R. Rastogi, "Load balancing of nodes in cloud using ant colony optimization," in *Computer Modeling and Simulation (UKSim)*, 2012 UKSim 14th International Conference on, pp. 3-8, IEEE, 2012.
- [21] S. G. Domanal and G. R. M. Reddy, "Load balancing in cloud computing using modified throttled algorithm," 2013.
- [22] S. Subramanian, G. Nitish Krishna, M. Kiran Kumar, P. Sreesh, and G. Karpagam, "An adaptive algorithm for

dynamic priority based virtual machine scheduling in cloud, "International Journal of Computer Science Issues (IJCSI), vol. 9, no. 6, 2012.

[23] B. Wickremasinghe, "Cloud analyst: A cloudsim-based tool for modeling and analysis of large scale cloud computing environments,"MEDC Project Report, vol. 22, no. 6, pp. 433-659, 2009.

[24] S. Mohapatra, S. Mohanty, and K. S. Rekha, "Analysis of different variants in round robin algorithms for load balancing in cloud computing,"International Journal of Computer Applications, vol. 69, no. 22, 2013.

[25] A. Gulati and R. K. Chopra, "Dynamic round robin for load balancing in a cloud computing," 2013.

[26] C.-C. Lin, P. Liu, and J.-J. Wu, "Energy-aware virtual machine dynamic provision and scheduling for cloud computing," in Cloud Computing (CLOUD), 2011 IEEE International Conference on, pp. 736-737, IEEE, 2011.

IJSER