LOAD BALANCING OF USER PROCESSES AMONG VIRTUAL MACHINES IN CLOUD COMPUTING ENVIRONMENT

[1]Neha Singla
Sant Longowal Institute of Engineering and Technology, Longowal, Punjab, India
Email:[1]neha.singla7@gmail.com

Abstract— Load balancing is a methodology to distribute load across many computers, or other resources over the network links to achieve good resource utilization, to decrease data processing time, minimum average response time, and minimize overload. The establishment of an effective load balancing algorithm and how to use Cloud computing resources efficiently for effective and efficient cloud computing is one of the ultimate goal. So in this, we explore the coordination between DC (Data Centers) and UB (user bound) to optimize the application performance and response time by using a tool called Cloud Analyst that can maintain the load balancing and provides better improved strategies through efficient job scheduling and modified resource allocation techniques.

Index Terms—Cloud Analyst, Load Balancing, Processing Time, Response Time, Virtualization.

I. INTRODUCTION

Cloud computing is a new type of computing mode. It distributes computation task on the resource pool which consists of large number of computers, so that the application systems can acquire the computation power, the space for storage and software service according to its demand. This kind of resource pool is called cloud. The Clouds are some virtual computation resources which can be maintained and managed by them, usually they are some large-scale server clusters, including calculating server, storage server, the broadband resources and so on.

In the cloud computing domain, all virtualization solutions are system integration solutions that include servers, systems for storage, network devices, software and service. They include many layers of virtualization technologies such as hardware virtualization, network virtualization, application virtualization and desktop virtualization, and combine several layers flexibly to perceive the different models of virtualization solutions according to the environment of application.

The key idea of load distribution here is to effectively and efficiently utilize the infrastructure available in the cloud for processing user's requests. An efficient load balancer greatly improves the efficiency and resource availability of the cloud. The load balancing problem here is to handle efficient distribution of user processes across virtual Machines running on host servers in the cloud. The load balancer chooses an appropriate VM from among the running VMs and redirects the user's request to that VM which will actually be responsible for handling the user's request.

II. RELATED WORK

In this section, we describe the related work of load balancing in cloud environment. The author of paper [1] gave brief explanation of security issues in Cloud Computing. Security has always been the main issue for IT Executives when it comes to cloud adoption. While the use of services simplifies resource use, the services themselves have to be discovered and selected so clients can make effective use of resources [2]. The author of [3] Proposed a new simulator known as CloudAnalyst. With the advancement of cloud technologies, it is necessary to simulate large scale applications on cloud before their real time implementation. Using simulation, [4] studied the behavior of cloud environment. CLOUDS Laboratory introduced Cloud-Analyst, the Cloud-Sim based tool. As there are many simulators such as Gridsim, Cloudbees, Cloudsim etc. which are used to perform operations in Cloud Computing Environment. Kim [5] describe simlutor Cloudsim which is an exclusive Cloud Computing simulator of Cloud Computing and Distributed Systems (CLOUDS) Laboratory at the University of Melbourne, Australia. [6] implemented a network model and also a memory model into Cloudsim simulator. The author [7] established an efficient load balancing algorithm to accomplish Cloud Computing goals and also for the usage of Cloud Computing resources efficiently. [8] discussed some of the Key technologies of Cloud Computing such as Virtualization technology, Security Management, Programming Model and Data Management. By using efficient load balancing techniques the authors [9] made an attempt to enhance dynamic cloud based services. The authors [10] defined cloud computing as a large-scale distributed computing paradigm that is driven by economies of scale, in which a pool of abstracted, virtualized, dynamically-scalable, managed computing power, storage, platforms, and services are delivered on demand to external customers over the Internet. The author [11] proposed CloudSim: an extensible simulation toolkit that enables modelling and simulation of Cloud computing systems and environments.

III. PROBLEM STATEMENT

Cloud computing is a term, which involves virtualization, distributed computing, networking, and software and web services. The main issue Central to the issues lies on the establishment of an effective load balancing algorithm. Load balancing is a computer networking method for distribution of workloads across various computers or a large number of computers, network links, central processing units, disk drives, or other resources. In order to successfully balancing the load optimizes resource use, exaggerate throughput, reduces response time, and evades overload. The main objectives are:

- Completely analyzation of CloudAnalyst Simulator.
- Effective utilization of resources.
- To distribute user processes efficiently on virtual machines running in the cloud.
- To considerably improve the performance.
- To entertain future modification in the system.

IV. METHODOLOGY

The Proposed methodology is used to balance the load of the user requests by using Cloud Analyst tool in round robin fashion. The load is balanced on Virtual Machine (VM) in order to achieve better response time and processing time. The load balancing is done before it reaches the processing servers. The job is scheduled based on various parameters like the speed of processor and accredited load of Virtual Machine (VM) and etc. It maintains the information in each VM and numbers of request currently allocated to VM of the system. It identify the least loaded machine and also the allocation of it when a request come and it identified the first one if there are more than one least loaded machine. In this case study, we model the behavior of social network applications such as Facebook by using CloudAnalyst to evaluate response time and data processing time. For simplicity each user base is contained within a single time zone and let us assume that most users use the application in the evenings after work for about 2 hours. Let us also assume that 5% of the registered users will be online during the peak time simultaneously and only one tenth of

that number during the off-peak hours. Let us also assume that each user makes a new request every 5 minutes when online.

The user requests are to be balanced on the Virtual Machine. Virtual machine enables the abstraction of an OS and Application running on it from the hardware. The infrastructure services of interior hardware interrelated to the Clouds is modeled in the Cloudsim simulator by a Datacenter element for handling service requests. These requests are application elements sandboxed within VMs that needs to be allocated a share of processing power on Datacenter's host components. DataCenter object handles 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.

The Data Center Controller uses a VmLoadBalancer to determine which VM should be assigned the next request for processing. The expected response time can be found with the help of the following formulas:

Response Time = Fint - Arrt + TDelay … (1)

Where, Arrt is the arrival time of user request

Fint is the finish time of user request

The transmission delay can be decisive by using the following formulas:

TDelay = T + T Latency transfer ……..…. (2)

Where, TDelay is the transmission delay

Tlatency is the network latency and

T transfer is the time taken to transfer the size of data of a single request (D) from source location to destination.

Ttransfer = D / Bwperuser ……. (3)

Bwperuser = Bwtotal / Nr …….. (4)

Where, Bwtotal is the total available bandwidth and

Nr is the number of requests from users that are currently in transmission.

V. EXPERIMENT AND EVALUATION

In our case study, we will consider 3 scenarios. All the three scenarios will be implemented by using the five steps that are explained in the previous section. The detail description is as follows.

Step 1: Creation of Cloud Environment

When CloudAnalyst is started the first screen displayed is the main screen. It consists of the simulation panel outline with a map of the world on the right and the main control panel on the left. It is shown by using figure.

Fig 5.1 CloudAnalyst Main Screen

Step 2: Configuration Setup: In this, we will define the User bases and Datacenters. The distribution of users is shown in the following table.

Table I: Distribution of Users

Region	CloudAnalyst Region Id	Users
North America	0	80 million
South America	1	20 million
Europe	2	60 million
Asia	3	27 million
Africa	4	5 million
Ocenia	5	8 million

Step 3: Creation of Virtual Environment: In this, the values will be assigned to userbases and datacenters.

Allocation of Userbases: The number of Userbases along with the regions in which they are defined shown in the following figure.

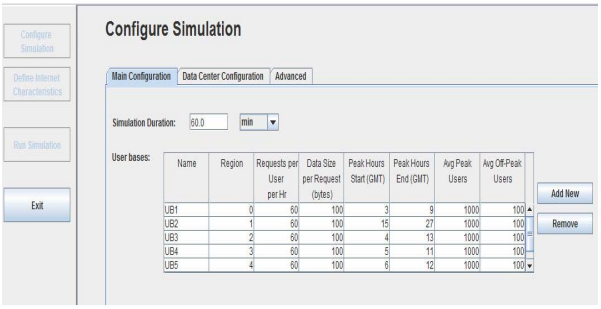

Fig 5.2 Configurations of Userbases

Allocation of Datacenters: The allocation of three datacenters with varying Virtual Machines shown in the following figure. Here the Virtual Machines, initially, will be 50 and it will be vary according to the scenarios.

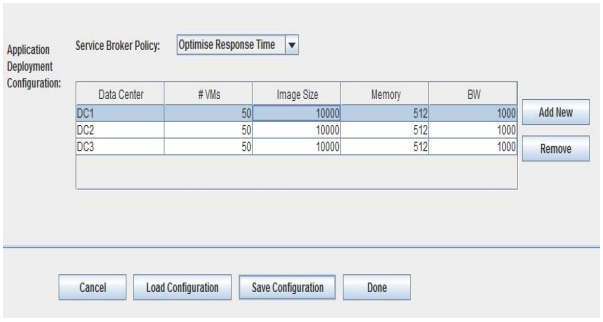

Fig 5.3 Allocations of Datacenters

Datacenter Configuration: The data center tab allows us to define the configuration of a data center. When we select a data center from this table a second table will appear below it with the details of the server machines in the data center shown as in the following figure.

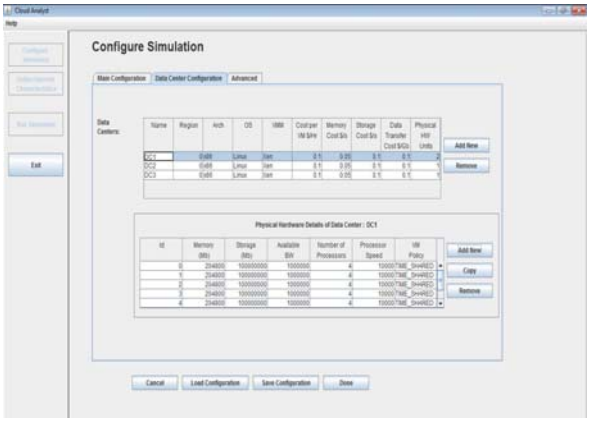

Fig 5.4 Configurations of Datacenters

Grouping Factors: It contains some important parameters that apply to the entire simulation explained as follows.

User Grouping Factor (in User Bases): This parameter tells the simulator how many users should be treated as a single bundle for generating traffic. The number inclined here will be used as the number of requests represented by a single InternetCloud

Request Grouping Factor (in Data Centers): This criteria tells the simulator how many requests should be treated as a single unit for processing. I.e. these innumerable requests are wrapped together and assigned to a single VM as a unit.

Executable instruction length (in bytes): It is the main criteria that affect the execution length of a request.

Load balancing policy: The load balancing approach used by all data centers in allocating requests to virtual machines. The relevant policies are:

a. Round-robin
b. Equally Spread Current Execution
c. Throttled

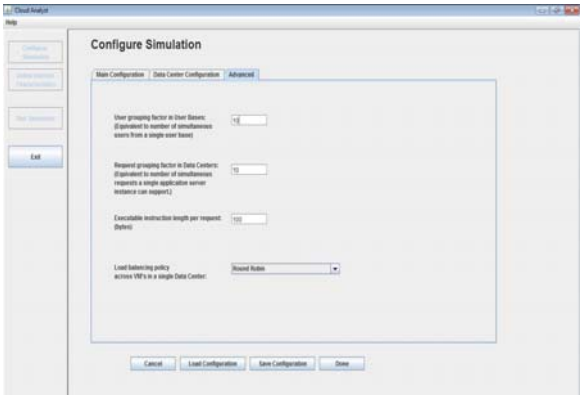

Fig 5.5 Configurations of Grouping Factor

Parameters Used : The Internet Characteristics screen can be used to set the Internet latency and bandwidth parameters. It presents two matrices for these two categories.

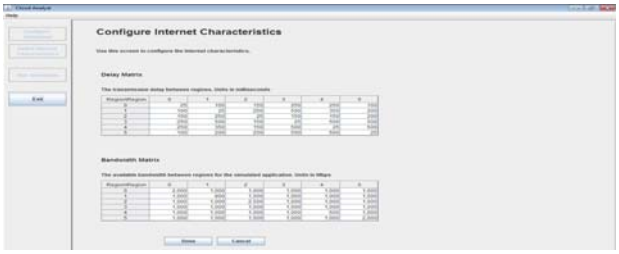

Fig 5.6 Internet Characteristics

The Cloud Analyst Internet is a consideration for the real world Internet, implementing only the features that are important to the simulation. It illustrates the traffic routing over the internet around the globe by introducing suitable transmission latency and data transfer delays. The transmission latency and the feasible bandwidth between the 6 regions are configurable.

Scenario 1: Web Application Hosted on 3 Data Centers with 50 VMs each

Like with real world applications, initially the application is deployed on three datacenters in Region 0 (North America), Region 1 (Europe), Region 2 (Asia) with 50 VMs each. We use Optimum Response Time with Round Robin VM load Balancing Policy. The number of users will vary.

Table II: Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	50.03	40.43	62.18
UB2	51.15	41.27	63.27
UB3	51.16	41.50	62.25
UB4	301.43	232.26	379.27
UB5	300.01	237.52	382.26
UB6	200.21	160.19	245.19

Form above Table II, we conclude that as the distance between Userbase and Datacenter increases the Response Time will also increase. For Userbase1 and Userbase6, the Response Time is less as it is near to the Datacenter but for the other Userbases, the Response Time increased with the increase in distance from datacenter.

Table III: Datacenter Request Servicing Timing

Datacenter	Avg (ms)	Min (ms)	Max (ms)
DC1	0.36	0.03	0.95
DC2	1.58	0.14	2.01
DC3	1.49	0.13	2.00

From above Table III, we conclude that Datacenter1 take less processing time and DC2, DC3 have almost same processing time.

Table IV: Overall Response Time

	Avg (ms)	Min (ms)	Max (ms)
Response Time	159.47	40.43	382.26
Data Center Processing Time	1.13	0.03	2.901

From above table IV, the result may not be quite what was expected by bringing the service closer to the users. The response time has roughly high because of high peak load traffic. The average response time is high. To improve the response time we will increase the number of VMs to transfer the load from heavily loaded to lesser loaded with the same memory, hardware and other parameters.

Scenario 2: Web Application Hosted on 3 Data Centers with 100 VMs each

Like with real world applications, initially the application is deployed on three datacenters in Region 0 (North America), Region 1 (Europe), Region 2 (Asia) with 100 VMs each. We use Optimum Response Time with Round Robin VM load Balancing Policy. The number of users will vary.

Table V: Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	50.17	40.61	62.36
UB2	52.38	43.53	64.52
UB3	52.37	42.75	63.50
UB4	302.79	233.51	380.52
UB5	301.09	238.02	383.51
UB6	52.38	42.51	63.76

Form above Table V, we conclude that with the increase in VMs as the distance between Userbase and Datacenter increases the Response Time will also increase. For Userbase1 and Userbase6, the Response Time is less as it is near to the Datacenter but for the other Userbases, the Response Time increased with the increase in distance from datacenter. Also, UB4 has high Response Time as it is far away from datacenter.

Table VI: Datacenter Request Servicing Timing

Datacenter	Avg (ms)	Min (ms)	Max (ms)
DC1	0.72	0.08	1.13
DC2	2.77	0.26	3.27
DC3	2.70	0.25	3.25

From above Table VI, we conclude that Datacenter1 take less processing time and DC2, DC3 have almost same processing time. Also if we compare it with Table 8.3, we can see that with the increase in the number of VMs the processing time of datacenters also increases.

Table VII: Overall Response Time

	Avg (ms)	Min (ms)	Max (ms)
Response Time	135.71	40.61	383.51
Data Center Processing Time	2.38	0.08	3.27

From above Table VII, we conclude that increasing the VMs will improve the response time but it will require more processing time.

Scenario 3: Web Application Hosted on 3 Data Centers with 1000 VMs each

Like with real world applications, initially the application is deployed on three datacenters in Region 0 (North America), Region 1 (Europe), Region 2 (Asia) with 1000 VMs each. We use Optimum Response Time with Round Robin VM load Balancing Policy. The number of users will vary.

Table VIII: Response Time by Region

Userbase	Avg (ms)	Min (ms)	Max (ms)
UB1	53.48	42.62	65.58
UB2	59.40	45.02	74.27
UB3	59.64	43.75	73.01
UB4	309.42	218.51	388.02
UB5	309.05	239.52	491.08
UB6	59.62	44.25	71.26

Now in this scenario, we have increased VMs 100 to 1000. From Table 8.8 we conclude that the response time of the Usebases is increased because of the memory and parameters are same. But the Criteria is same as Userbase is moving away from Datacenter the response time will also increase.

Table IX: Datacenter Request Servicing Timing

Datacenter	Avg (ms)	Min (ms)	Max (ms)
DC1	3.79	0.36	4.33
DC2	9.93	1.01	10.77
DC3	9.98	1.00	10.76

From Table IX, we conclude that by increasing the VMs on the same parameters defined for the other scenarios, the processing time also increases.

Table X: Overall Response Time

	Avg (ms)	Min (ms)	Max (ms)
Response Time	141.92	42.62	491.08
Data Center Processing Time	8.87	0.36	10.77

From Table X, we conclude that if we increase the VMs on the same memory then the Response time will be improved but only from Scenario 1. The response time is increased as compare to the response time for the VMs 100. If we want to improve the response time then we have to increase the memory and other parameters also. The processing time will also increase.

VI. RESULTS AND DISCUSSIONS

After performing the number of experiments, the result summary of the response time and processing time can be tabularized as follows

Average Response Time: From Table IV, Table VII, Table X, the average response time of all the three cases shown below.

Table XI: Average Response Time

	50 VMs	100VMs	1000VMs
Response time	159.47	135.71	141.92

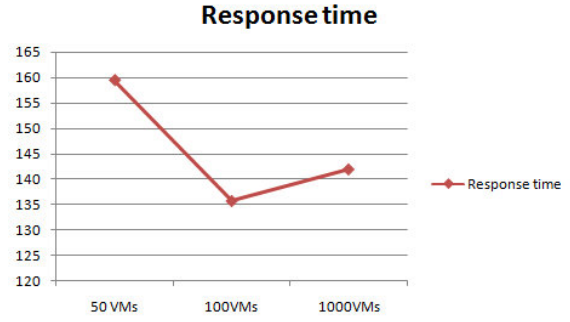

Fig 6.1: Average Response Time

From the above graph we conclude that the response time is high during the heavy peak load with 50 VMs. But if we increase the number of VMs on the same memory parameters then the response time will be improved but for the case of 1000 VMs the response time will not improve as much. To improve the response time we have to increase the memory and other parameters.

Average Data Processing Time: From Table IV, Table VII, Table X, the average data processing time of all the three cases shown below.

Table XII: Average Data Processing Time

	50 VMs	100VMs	1000VMs
Processing time	1.13	2.38	8.87

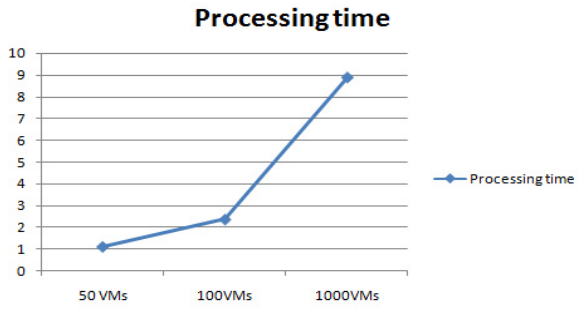

Fig 6.2: Data Processing Time

From the above graph we conclude that with the increase in the number of VMs the data processing time will also increased.

VII. CONCLUSION

Cloud Computing is a vast concept and load balancing plays a very important role in case of Clouds. There is an immense scope of improvement in this area. We have discussed only load scheduling by using round robin that can be applied to clouds to improve the response time and processing time, but there are still other approaches that can be applied to balance the load in clouds. The performance can also be increased by varying different parameters.

VIII. REFERENCES

[1] Aderemi A. Atayero and Oluwaseyi Feyisetan, "Security Issues in Cloud Computing, the Potentials of Homomorphic Encryption," Journal of Emerging Trends in Computing and Information Sciences, Vol. 2 pp. 546-552, 2011.

[2] Andrzej Goscinski and Michael Brock, "Toward Dynamic and Attribute Based Publication, Discovery and Selection for Cloud Computing," Journal of Future Generation Computer Systems, pp. 947-970, 2011.

[3] Anton Beloglazov, Jemal Abawajy and Rajkumar Buyya, "Energy-Aware Resource Allocation Heuristics for Efficient Management of Datacenters for Cloud Computing," Journal of Future Generation Computer Systems, pp. 755-768, 2012.

[4] Dhaval Limbani and Bhavesh Oza, "A Proposed Service Broker Strategy in CloudAnalyst for Cost-Effective Data Center Selection," In Proceedings of International Journal of Engineering Research and Applications (IJERA), Vol. 2, pp.793-797, 2012.

[5] Hadi Salimi, Mahsa Najafzadeh and Mohsen Sharifi, "Advantages, Challenges and Optimizations of Virtual Machines Scheduling in Cloud Computing Environments," In Proceedings of International Journal of Computer Theory and Engineering, Vol. 4, pp. 189-193, 2012.

[6] Ibrahim Takouna, Wesam Dawoud, and Christoph Meinel, "Analysis and Simulation of HPC Applications in Virtualized Data Centers," In Proceedings of IEEE International Conference on Green Computing and Communications, Conference on Internet of Things, and Conference on Cyber, Physical and Social Computing, pp. 498-507, 2012.

[7] Jasmin James and Dr. Bhupendra Verma, "Efficient VM Load Balancing Algorithm for a Cloud Computing Environment," In Proceedings of International Journal on Computer Science and Engineering (IJCSE), Vol. 4, pp. 1658-1663, 2012.

[8] Juefu Liu and Peng Liu, "Status and Key Technologies in Cloud Computing," In proceedings of 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE), pp. 285-288, 2010.

[9] Karthik V Bellur, Krupal M, Praveen Jain and Dr.Prakash Raghavendra, "Achieving Operational Efficiency With Cloud Based Services," In Sixth International Conference on Computer Science & Education (ICCSE 2011), pp. 1063-1068, 2011.

[10] Loganayagi.B and S.Sujathaa, "Enhanced Cloud Security by Combining Virtualization and Policy Monitoring Techniques," In Proceedings of International Conference on Communication Technology and System Design, pp. 654-661, 2011.

[11] Rodrigo N. Calheiros, Rajiv Ranjan, Anton Beloglazov, César A. F. De Rose, and Rajkumar Buyya, "CloudSim: A Toolkit for the Modeling and Simulation of Cloud Resource Management and Application Provisioning Techniques", 2011.