Simulation of Load Balancing Algorithms using Cloud Analyst

Azizkhan F Pathan
MTech (Computer Science and Engineering)
Bapuji Institute of Engineering and Technology
Davangere, India
apathan21@gmail.com

Sneha.N
MTech (Computer Science and Engineering)
Bapuji Institute of Engineering and Technology
Davangere, India
sneha2442@gmail.com

***Abstract*— Cloud computing today has now been rising as new technologies and new business models. Cloud computing is spreading globally, due to its easy and simple service oriented model. The increasing cloud computing services offer great opportunities for clients to find the maximum service and finest pricing, which however raises new challenges on how to select the best service out of the huge group. Cloud computing employs a variety of computing resources to facilitate the execution of large-scale tasks. Therefore, to select appropriate node for executing a task is able to enhance the performance of large-scale cloud computing environment. It is time-consuming for consumers to collect the necessary information and analyze all service providers to make the decision. This is also a highly demanding task from a computational perspective, because the same computations may be conducted repeatedly by multiple consumers who have similar requirements. Also the numbers of users accessing the cloud are rising day by day. As the cloud is made up of datacenters; which are very much powerful to handle large numbers of users still then the essentiality of load balancing is vital. Load balancing is the process of distributing the load among various nodes of a distributed system to improve both resource utilization and job response time while also avoiding a situation where some of the nodes are heavily loaded while other nodes are idle or doing very little work. Cloud Analyst is a tool that helps developers to simulate large-scale Cloud applications with the purpose of understanding performance of such applications under various deployment configurations. The simulated results provided in this paper are based on the scheduling algorithms Round Robin, Equally Spread Current Execution and Throttled load balancing policies. The aim of this paper is to briefly discuss about various efficient and enhanced load balancing algorithms and experimentally verify how to minimize the response time and processing time through the tool called cloud analyst.**

Keywords—Cloud computing; Load balancing; Round Robin; Equally Spread Current Execution; Throttled; Response time minimization; Cloud Analyst.

I. INTRODUCTION

Cloud computing is an attracting technology in the field of computer science. In Gartner's report [1], it says that the cloud will bring changes to the IT industry. The cloud is changing our life by providing users with new types of services. Users get service from a cloud without paying attention to the details [2]. NIST gave a definition of cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources(e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction[3]. Cloud computing is an evolving paradigm with changing definitions, it is defined as a virtual infrastructure which provides shared information and communication technology services, via an internet i.e. cloud. Cloud computing provides a computer user access to Information Technology (IT) services (i.e., applications, servers, data storage) without requiring an understanding of the technology or even ownership of the infrastructure. Cloud Computing is getting advanced day by day. Cloud works on the principle of virtualization of resources with on-demand and pay-as–you go model policy.

The five important characteristics of cloud computing defined by NIST includes On-demand self-service, Global network access, Distributed resource pooling, Scalable , Measured service[4].

Based on the domain or environment in which clouds are used, clouds can be divided into 3 categories Public Clouds (which are available to clients from a third party service provider via the Internet. e.g., Amazon ,Google and IBM),Private Clouds(a business essentially turns its IT environment into a cloud and uses it to deliver services to the users),Hybrid Clouds(Hybrid cloud means either two separate clouds joined together (public, private, internal or external or a combination of virtualized cloud server instances used together with real physical hardware)[5].

In the whole, cloud computing provides us the attracting conventional services like: Software as a Service (SaaS) ,where the user uses different software applications from different servers through the Internet which can be consumed using web browsers without purchasing or maintaining overhead. Some examples are Gmail, Salesforce.com, etc. Platform as a Service (PaaS), where Application development framework offered as a service to developers for quick deployment of their code. Some examples for PaaS include Google App Engine, Heroku, Cloud Foundry, etc. Last but not the least Infrastructure as a Service (IaaS) where Shared infrastructure such as servers, storage and network are delivered as a service over the internet. Some examples include Amazon Web Services, Rackspace Cloud, etc[6].

Cloud computing architectures are inherently parallel, distributed and serve the needs of multiple clients in different scenarios. This distributed architecture deploys resources distributive to deliver services efficiently to users in different

geographical channels [7].Clients in a distributed environment generate request randomly in any processor. So the major drawback of this randomness is associated with task assignment. The unequal task assignment to the processor creates imbalance i.e., some of the processors are overloaded and some of them are under loaded [8]. The objective of load balancing is to achieve a high user satisfaction and resource utilization ratio, and to avoid the situation where nodes are either heavily loaded or under loaded in the network, hence improving the overall performance of the system. Proper load balancing can help in utilizing the available resources optimally, thereby minimizing the resource consumption [9].

There are mainly two types of load balancing algorithms .In Static algorithm the traffic is divided evenly among the servers. This algorithm requires a prior knowledge of system resources, so that the decision of shifting of the load does not depend on the current state of system. Static algorithm is proper in the system which has low variation in load. In Dynamic algorithm the lightest server in the whole network or system is searched and preferred for balancing a load. For this real time communication with network is needed which can increase the traffic in the system. Here current state of the system is used to make decisions to manage the load [10].

Cloud computing implements virtualization technique in which a single system can be virtualized into number of virtual systems [11]. Load balancing decides which client will use the virtual machine and which requesting machines will be put on hold. Load balancing of the entire system can be handled dynamically by using virtualization technology where it becomes possible to remap Virtual Machines (VMs) and physical resources according to the change in load. Due to these advantages, virtualization technology is being comprehensively implemented in Cloud computing. A Virtual Machine (VM) is a software implementation of a computing environment in which an operating system (OS) or program can be installed and run. The Virtual Machine typically changes a physical computing environment and requests for CPU, memory, hard disk, network and other hardware resources are managed by a virtualization layer which translates these requests to the underlying physical hardware. VMs are created within a virtualization layer, such as a hypervisor or a virtualization platform that runs on top of a client or server operating system. This operating system is known as the host OS. The virtualization layer can be used to create many individual, isolated VM environments, where multiple requests or tasks can execute in multiple machines [12].

II. DYNAMIC LOAD BALANCING ALGORITHMS

A. *Round Robin Algorithm*

The name of this algorithm suggests that it works in round robin manner [13]. When the Data Center Controller gets a request from a client it notifies the round robin load balancer to allocate a new virtual machine (VM) for processing. Round robin load balancer (RRLB) picks a VM randomly from the group and returns the VM id to Data Center Controller for processing. In this way the subsequent requests are processed in a circular order. However there is a better allocation policy called weighted round robin balancer [14] in which we can assign a weight to each VM so that if one VM is capable of handling twice as much load as other then the former gets the weight of 2 whereas the later gets the weight of 1.

B. *Active Monitoring Load Balancing Algorithm*

This algorithm is also called as equally spread current execution (ESCE) load balancing. It uses active monitoring load balancer for equally spreading the execution of loads on different virtual machines. The steps of this algorithm are described as follows referring to Fig 1. Active monitoring load balancer (AMLB) maintains an index table of virtual machines and the number of allocations assigned to each virtual machine. Data Center Controller receives a new request from a client. When a request for allocation of new VM from Data Center Controller arrives at AMLB, it parses the index table from top until the least loaded VM is found. When it finds, it returns the VM id to the Data Center Controller. If there is more than one found, AMLB uses first come first serve (FCFS) basis to choose the least loaded. Simultaneously, it also returns the VM id to the Data Center Controller. Then the Data Center communicates the VM identified by that id. The Data Center Controller notifies the AMLB about the new allocation. After that AMLB updates the allocation table by increasing the allocation count by 1 for that VM. When a VM suitably finishes processing the assigned request, it forwards a response to the Data Center Controller. On receiving the response it notifies the AMLB about the VM de-allocation The AMLB updates the allocation table by decreasing the allocation count for that VM by 1.

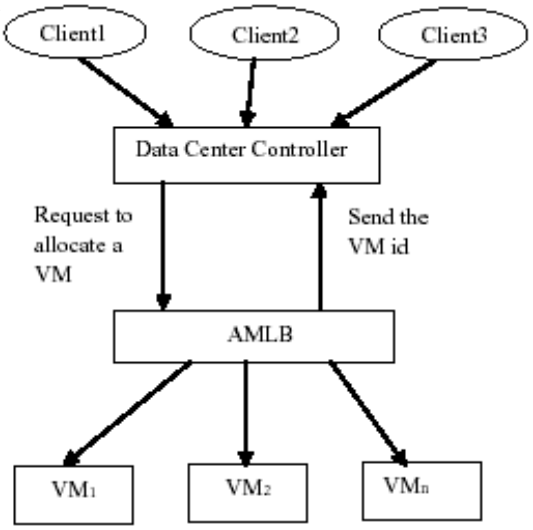

Figure 1.Active monitoring load balancing.

C. *Throttled Algorithm*

This algorithm implements a throttled load balancer (TLB) to monitor the loads on each VM. Here each VM is assigned to only one task at a time and can be assigned another task only when the current task has completed successfully. The algorithm steps can be described as follows: The job of TLB is to maintain an index table of all VMs as well as their current states (Available or Busy). The client first makes a request to Data Centre Controller for the allocation of appropriate VM and to perform the recommended job. The Data Centre Controller queries the TLB for allocation of the VM. The TLB scans the index table from top to bottom until the first

available VM is found. If it finds, then TLB returns the VM id to the Data Center Controller. The Data Centre communicates the request to the VM identified by the id. Further, the Data Centre acknowledges TLB about the new allocation and revises the index table by increasing the allocation for that VM by 1. On the other hand, if the TLB doesn't find any VM in the available state it simply returns null. In this case Data Center Controller queues the request until the availability of any VM. When a VM suitably finishes processing the request, it forwards a response to the Data Center Controller. On receiving it, the Data Center Controller acknowledges the TLB regarding VM de-allocation. The TLB updates the allocation table by decreasing the allocation count for the VM by 1[15].

III. RESPONSE TIME CALCULATION

The purpose of these algorithms is to calculate the expected response time. We use the following formula for calculation:

$$\text{Response Time} = \text{Fint} - \text{Arrt} + \text{Tdelay}$$

where, Arrt is the arrival time of user request and Fint is the finish time of user request and Tdelay is the transmission delay. However, Tdelay can be calculated as:

$$\text{Tdelay} = \text{Tlatency} + \text{Ttransfer}$$

Here, Tlatency is the network latency and Ttransfer is the time taken to transfer the size of data of a single request (D) from source location to destination.Tlatency is taken from the latency matrix (after applying Poisson distribution on it for distributing it)held in the internet characteristics.

$$\text{Ttransfer} = D / \text{Bwperuser}$$
$$\text{where Bwperuser} = \text{Bwtotal} / N;$$

Bwtotal is the total available bandwidth (held in the internet characteristics) and N is the number of user requests currently in transmission. The internet characteristics also keep track of the number of user requests in-flight between two regions for the value of N.

IV. SIMULATION STEP

In order to analyze the above discussed algorithms we have used the tool called Cloud Analyst [16]. Basically cloud analyst is a cloudsim [17] based GUI tool used for modeling and analysis of large scale cloud computing environment. Moreover, it enables the modeler to execute the simulation repeatedly with the modifications to the parameters quickly and easily. The following diagram shows the GUI interface of cloud analyst tool.

It comes with three important menus: configure simulation, define internet characteristics and run simulation [17] [18].These menus are used for setting of the entire simulation process.

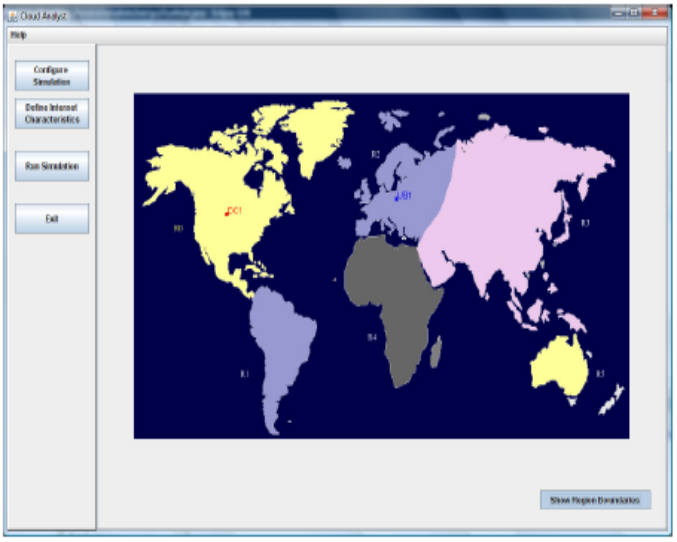

Figure2. GUI Interface of Cloud Analyst.

The tool provides us the feature of switching algorithms according to our requirement. Simulation setup and analysis of results are carried out for a period of 60 minutes by taking different numbers of users, 3 data centers i.e. DC1, DC2, and DC3 having 75, 50 and 25 numbers of VMs respectively. The other parameters are fixed according to Table 1 as shown.

TABLE 1: Setting of Parameters

Parameter	**Value passed**
VM-image size	10000
VM-memory	1024MB
VM-Bandwidth	1000
Service Broker Policy	Opimise response time
Data center architecture	x86
Data center-OS	Linux
Data center-VMM	Xen
Data center-No of VMs	DC1-75 DC2-25 DC3-50
Data center-memory per machine	2GB
Data center-storage per machine	1GB
Data center-available bandwidth per machine	1000000
Data center-processor speed	1000
Data center-VM policy	Time shared
User grouping factor	1000
Request grouping factor	250
Executable instruction length	250

V. EXPERIMENTAL RESULTS

After performing the simulation, the results computed by cloud analyst is as shown in the tables given below. We have used the above defined configuration for each load balancing policy one by one and depending on that the result calculated for the metrics like response time, request processing time in fulfilling the request has been shown. Overall response time calculated by the cloud analyst for each loading policy has been shown in the table 2, 3 and 4 respectively. As can be seen from the figure the overall response time of Round Robin policy and ESCEL policy is almost same while that of the Throttled Policy is somewhat low as compared to other two policies.

TABLE 2: Response time using Round Robin Policy

	Overall Response time	**Datacenter processing time**
Avg(ms)	187.78	29.13
Min(ms)	55.85	12.51
Max(ms)	384.61	61.69

TABLE 3: Response time using Active Monitoring Load Balancing Policy

	Overall Response time	**Datacenter processing time**
Avg(ms)	187.72	29.13
Min(ms)	55.85	12.51
Max(ms)	384.61	61.69

TABLE 4: Response time using Throttled Policy

	Overall Response time	**Datacenter processing time**
Avg(ms)	187.68	29.13
Min(ms)	55.85	12.51
Max(ms)	381.96	61.69

From the table it is clear that average processing time at datacenters for all three policies is same.

CONCLUSION

The response time and data transfer cost is a challenge of every engineer to develop the products that can increase the business performance in the cloud based sector. The several strategies lack efficient scheduling and load balancing resource allocation techniques leading to increased operational cost and give less customer satisfaction. A greater challenge in minimization of response time is widely seen for each and every engineer of IT sector to develop the products which can increase the efficiency of business performance, customer satisfaction in cloud based environment. Keeping these things in mind we have simulated three different load balancing algorithms: Round robin, Active monitoring and Throttled for executing the user requests in cloud environment. Each algorithm is observed and their scheduling criteria like average response time and data center processing time are found.

We have found that the parameters: response time, data processing time are almost similar in case of Round Robin and active monitoring load balancing policies. However, these parameters are slightly improved in case of Throttled load balancing. Hence we conclude that Throttled load balancing is an effective and efficient one than the other two that discussed. The performance of the given algorithms can also be increased by varying different parameters or developing an innovative algorithm for provisioning of cloud computing resources. Future work can be based on this algorithm modified and implemented for real time system. Better response time can be expected if we apply some evolutionary algorithms such as PSO, ACO, and ABC instead of classical algorithms.

REFERENCES

[1] R. Hunter, The why of cloud, http://www.gartner.com /DisplayDocument?doc cd=226469&ref= g noreg, 2012.

[2] M. D. Dikaiakos, D. Katsaros, P. Mehra, G. Pallis,and A. Vakali, Cloud computing: Distributed internet computing for IT and scientific research, Internet Computing, vol.13, no.5, pp.10-13, Sept.-Oct. 2009.

[3] P. Mell and T. Grance, The NIST definition of cloud computing, http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf,2012. techtarget.com /definition/public-cloud, 2012.

[4] Mell,P, Grance, T "The NIST definition of Cloud Computing" version 15. National Institute of Standards and Technology (NIST),Information Technology Laboratory (October 7, 2009)

[5] AlexaHuth and JamesCebula"The Basics of Cloud Computing",www.us-cert.gov/reading.../USCERT CloudComputing HuthCebula."

[6] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal, James Broberg, and Ivona Brandic, "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility, Future Generation Computer Systems", Volume 25, Number 6, Pages: 599-616, ISSN: 0167-739X, Elsevier Science, Amsterdam, The Netherlands, June 2009.

[7] 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.

[8] A. Khiyaita, H. El Bakkli, M. Zbakh ,Dafir El Kettani," Load Balancing Cloud Computing: State Of Art", 2010,IEEE.

[9] 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.

[10] Load Balancing in Cloud computing, http://community.citrix.com /display/ cdn/ Load+Balancing.

[11] 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.

[12] 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.

[13] Saroj Hiranwal and Dr. K.C. Roy, "Adaptive Round Robin Scheduling Using Shortest Burst Approach Based On Smart Time Slice" International Journal Of Computer Science And Communication July-December 2011 ,Vol. 2, No. 2 , Pp. 319-323.

[14] Jasmin James and Dr. Bhupendra Verma, "Efficient VM load balancing algorithm for a cloud computing environment", International Journal on Computer Science and Engineering (IJCSE), 09 Sep 2012.

[15] Soumya Ranjan Jena and Zulfikhar Ahmad, " Response Time Minimization of Different Load Balancing Algorithms in Cloud Computing Environment", International Journal of Computer Applications, Volume 69,No.17,Pages:22-23 May 2013.

[16] Bhathiya Wickremasinghe, "CloudAnalyst: A CloudSim based Tool for Modelling and Analysis of Large Scale Cloud Computing Environments" MEDC project report, 433-659 Distributed Computing project, CSSE department., University of Melbourne, 2009.

[17] R. Buyya, R. Ranjan, and R. N. Calheiros, "Modeling And Simulation Of Scalable Cloud Computing Environments And The Cloudsim Toolkit: Challenges And Opportunities," Proc. Of The 7th High Performance Computing And Simulation Conference (HPCS 09), IEEE Computer Society, June 2009.

[18] Judith Hurwitz, Robin Bloor, and Marcia Kaufman, "Cloud computing for dummies" Wiley Publication (2010).