Load Balancing in Cloud Computing

Sumana R[1], Srinivas Achar[2]
Department of Computer Science Engineering
[1] Student, Atria Institute of Technology, Bangalore, India
[2] Assistant Professor, Atria Institute of Technology, Bangalore, India

Abstract- *Cloud Computing is the emerging internet based on technology which emphasizes commercial computing. Load Balancing helps in improving the execution of the unified server. In the present work, different calculations are dissected utilizing an using an analysis tool, namely, cloud analyst. Extensive amounts of information are created and traded over the system which further requires the need of increasingly registering assets. Cloud has helped ventures influence the advantages of processing assets which are shared over a virtualized domain. A great deal of ventures are now utilizing cloud-based administrations in either structure. This conveys us to the idea of load balancing in cloud.*

Keywords- cloud computing; load balancing; cloud analyst simulation; round robin algorithm; AMLB algorithm; throttled load balancing algorithm.

I. INTRODUCTION

Cloud computing is an innovation that trades information and give a great deal of assets to clients. Clients pay just for the assets that they utilized. Cloud computing stores information and disseminated assets in an open situation, and the measure of information stockpiling increments in all respects rapidly. In this way, load adjusting is the principle task in the cloud condition. Burden adjusting conveys the dynamic remaining task at hand over numerous hubs to guarantee that no hub is over-burden.

Cloud computing innovation is indicating exceptional development because of the headway of correspondence innovation, unstable utilization of Web and tackle extensive scale issues. Cloud computing is the idea of a "haze of figuring", as per which programs are propelled and yield the aftereffects of work in a standard internet browser window on a neighborhood PC, with all applications and their information important for activity situated on a remote server on the Web. The upsides of distributed computing incorporate the accompanying: decreased necessities for the figuring intensity of PCs, expanded adaptation to non-critical failure and security, the speed of information preparing increments ordinarily, costs for equipment and programming, for upkeep, control are diminished, and circle space is spared.

This study mainly focuses on analyzing the performance of cloud computing and comparing various load balancing algorithms using the Cloud Analyst network simulator.

II. LOAD BALANCING

Load balancing is utilized to circulate more burden to littler handling nodes to improve by and large framework execution [1]. In a distributed computing condition, load adjusting requires circulating the dynamic load equally between all nodes. Load balancing helps in the reasonable allotment of processing assets to accomplish an abnormal state of client fulfillment and appropriate utilization of assets. High asset usage and appropriate load balancing helps limit asset utilization. This executes adaptation to non-critical failure, versatility and maintain a strategic distance from troubles [2]. Load balancing is a technique that has assisted systems and assets, with providing greatest throughput with negligible reaction time. Load balancing is performed at two dimensions in cloud computing:

- The dimension of the virtual machine, the mapping is made between applications that are stacked in the cloud on the virtual machine. The node balancer allots the mentioned virtual machine to physical PCs, which adjusts the load of numerous applications from the PC.
- A host level, a mapping between the virtual machine and host assets that permit handling of a few approaching application demands.

III. EXISTING LOAD BALANCING ALGORITHMS

There are various load balancing algorithms used in cloud computing. In this study, the following three algorithms have been studied, which can be implemented in the Cloud Analyst simulator [4].

A. Round Robin Algorithm (RR)

This is the simplest algorithm that uses the concept of a quantum of time or interval (Fig. 1).

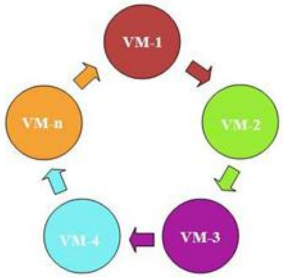


Fig. 1. Round Robin Algorithm (RR)

Here time is isolated into a few divisions, and every node is given a time quantum or time interval, and in this quantum the node will play out its tasks.

In Round Robin, booking a period quantum assumes an imperative job, supposing that the time cut is extremely huge, at that point the Round Robin booking calculation is equivalent to the FCFS arranging [4].

The disservice of the strategy is that, in spite of the fact that the calculation is extremely straightforward, yet to decide the quantum measure, it produces an extra burden on the scheduler. What's more, it has higher setting switches that expansion the turnaround time, and low throughput.

B. Active Load Balancing Monitoring (AMLB)

This calculation has a dynamic character. It stores data about each VM virtual machine and the quantity of solicitations that are as of now relegated to each VM. At the point when the solicitation is circulated by the new VM and if there are a few VMs, the principal remembered one is chosen, and the AMLB restores the VM identifier to the server farm controller. The data center controller warns the AMLB about the new distribution and sends the request to the virtual machine known under this VM identifier (Fig. 2).

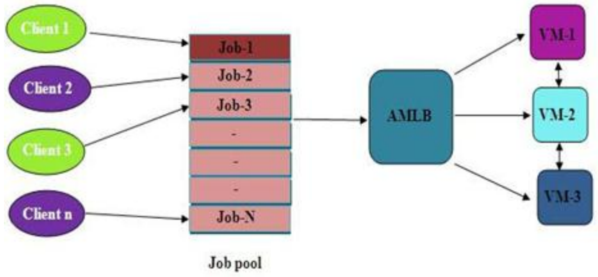


Fig. 2. AMLB Algorithm (AMLB).

The inconvenience of the calculation is that AMLB dependably finds the least stacked VM to dole out another approaching solicitation, however, does not check whether it was utilized before or not (accordingly some VM is utilized seriously, and some are yet not included).

C. Throttled Load Balancing Algorithm (TLB)

In this calculation, the node balancer keeps up a table of virtual machine files, just as their states (Accessible or Occupied). The client first makes a solicitation to the data center to locate an appropriate virtual machine (VM) to play out the suggested errand.

The data center demands a node balancer to circulate the virtual machine. The node balancer checks the record table from above until the most readily accessible virtual machine is found or the list table is totally filtered.

On the off chance that a virtual machine is discovered, the data center passes the solicitation to the virtual machine distinguished by the identifier. Furthermore, the data center affirms the node adjusting of the new dissemination, and the data center suitably changes the file table.

When processing a client request, if the corresponding VM is not found, the load balancer returns "-1" to the data center. The center request is processed by the data center.

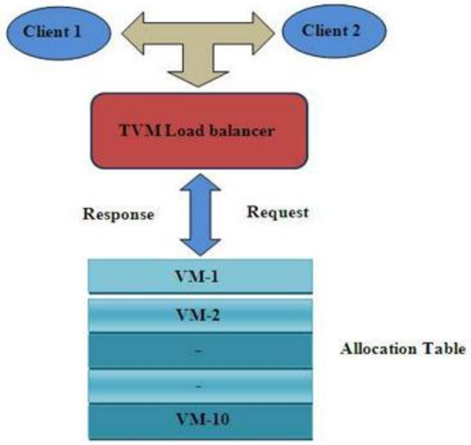


Fig. 3. Throttled Algorithm (TLB).

IV. CLOUD ANALYST SIMULATOR

The reenactment and investigation of the execution of the three burden adjusting calculations are performed utilizing the "Cloud Analyst" tool [5]. It enables the client to run different reenactments with little parameter changes, and furthermore enables you to redo the area of the clients who make the application and the area of the server farms [6]. Let's indicate the terminology of the emulator (Fig. 4):

– Region: in Cloud Analyst, the world is divided into 6 regions that coincide with the 6 major continents in the world;

– User Base: User Base is considered as a single unit, and is used to generate traffic;

– Data Processing Center: brokerage services determine which center should accept and process the request that comes from each user database;

– VmLoadBalancer: it is responsible for distributing the load to the available data center. VmLoadBalancer distributes the load in the data center based on the load balancing policy.

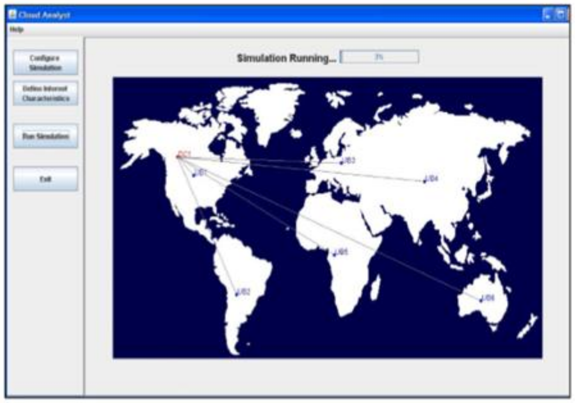


Fig. 4. Cloud Analyst Simulator.

In the modeling process, CloudSim 4.0 software was used.

V. SIMULATION AND EXPERIMENT

Reproduction and virtual test are the most ideal approach to test the calculation in cloud computing. Consider the work of each of the three load balancing algorithms using the example of the social network Facebook, which has more than 200 million registered users around the globe (Table 1).

TABLE I. REGISTERED USERS OF FACEBOOK AROUND THE WORLD

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

For the modeling, suppose we have a similar system, but on a scale of 1/10. Define 6 user databases representing the above 6 regions, with the following parameters (Table 2).

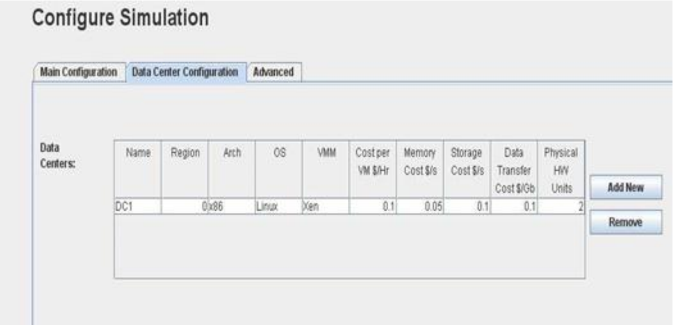
TABLE II. USER DATABASE SETTINGS

Base	Reg ion	Timezone	Peak hour (Local time)	Peak hour (GMT)	Users online during peak hours	Users online in non-peak hours
UB1	0	GMT – 6.00	7.00–9.00 pm	13:00-15:00	400,000	40,000
UB2	1	GMT – 4.00	7.00–9.00 pm	15:00-17:00	100,000	10,000
UB3	2	GMT + 1.00	7.00–9.00 pm	20:00-22:00	300,000	30,000
UB4	3	GMT + 6.00	7.00–9.00 pm	01:00-03:00	150,000	15,000
UB5	4	GMT + 2.00	7.00–9.00 pm	21:00-23:00	50,000	5,000
UB6	5	GMT+10.00	7.00–9.00 pm	09:00-11:00	80,000	8,000

We also define the data processing center (Table 3), which must process the request coming from each user database with the following parameters (Fig. 5).

TABLE III. DATA CENTER SETTINGS

Name	Dc1
Region	0
Arch	X86
OS	Linux
VMM	Xen
Cost per VM $/Hr	0.1
Memory Cost $/s	0.05
Storage Cost $/s	0.1
Data Transfer Cost $/GB	0.1
Physical Hw Units	20


Configure Simulation

Main Configuration | Data Center Configuration | Advanced

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

Fig. 5. Configuration of the data center.

Point of confinement the model to the way that every client database is contained in one time zone and expect that most clients utilize the application in the nights after work around 2 hours.

Assume additionally that 5% of enlisted clients will be on-line at pinnacle time all the while and one and only tenth of this number will be on the system amid off-top hours. Assume that every client makes another solicitation like clockwork, when on-line.

VI. SIMULATION RESULTS

We played out the reproduction multiple times as per the past parameters. Each time, we changed the node adjusting calculation, which was examined. The outcomes were looked at by the criteria: total response time (Figure 6), data center time (Figure 7), hourly data center load (Figure 8-10), and processing costs (Table 4).

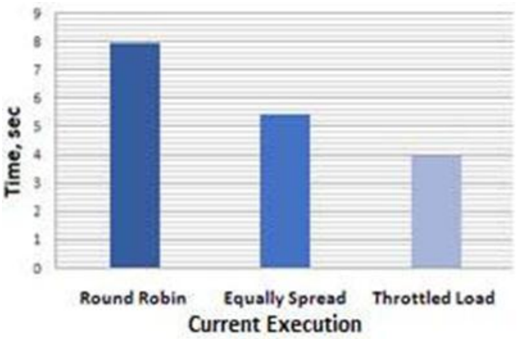


Fig. 6. Total response time.

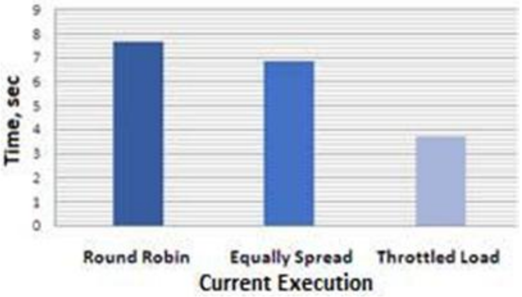


Fig. 7. Data center processing time

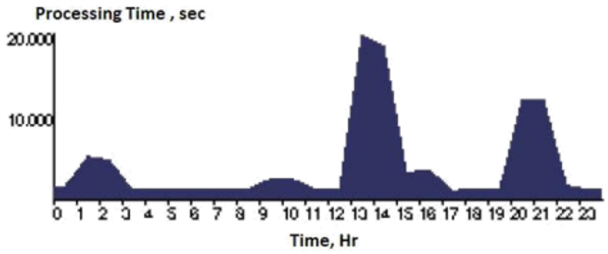


Fig. 8. Hourly data center load in (RR).

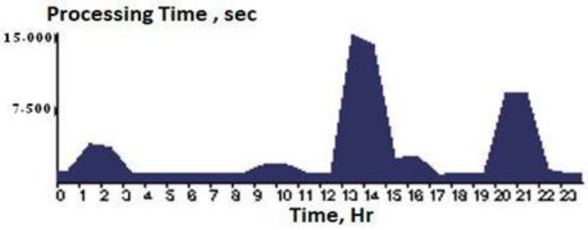


Fig. 9. Hourly data center load in (AMLB).

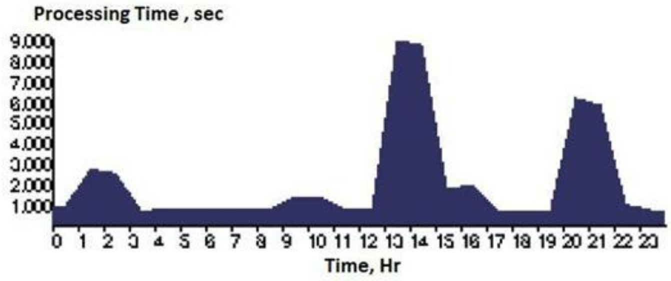


Fig. 10. Hourly data center load in (TLB).

TABLE IV. COST OF PROCESSING

Algorithm	VM Cost , $	Full Value , $
RR	120.5	632.34
AMLB	112.8	580.67
TLB	99.1	501.92

From the above review of load balancing and the three existing policies for the Cloud Analyst simulator, you can conclude that load balancing is a complex task in cloud computing.

Comparing the results obtained using different load balancing algorithms, we can conclude that the overall response time in the Throttled algorithm is better than in other algorithms, and the data center time is also better.

REFERENCES

[1] Buyya R., Ranjan R. and Calheiros R.N. Modeling and simulation of scalable cloud computing environments and the cloudsim toolkit: challenges and opportunities, High Performance Computing & Simulation HPCS'09, 2009, pp. 1-11.

[2] Desyatirikova E. N., Kuripta O. V. Quality management in IT service management based on statistical aggregation and decomposition

[3] approach, 2017 International Conference "Quality Management,Transport and Information Security, Information Technologies" (IT&QM&IS) , 2017, pp. 500-505. DOI: 10.1109/ITMQIS.2017.8085871.

[4] Calheiros R.N. CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services, Eprint: Australia, 2009, pp.9–17.

[5] Simar P.S. , Anju S. and Rajesh K. Analysis of load balancing algorithms using cloud analyst, International Journal of Grid and Distributed Computing, vol. 9, No. 9, 2016, pp.11-24.

[6] Maguluri S.T., Srikant R. and Ying L. Stochastic models of load balancing and scheduling in cloud computing clusters, in: INFOCOM Proceedings IEEE, 2012, pp. 702–710.

[7] Bhathiya W.A. CloudAnalyst: A CloudSim based Tool for Modelling and Analysis of Large Scale Cloud Computing Environments. University of Melbourne: Australia, 2009, p.21-35.