Load Balancing in Cloud Computing

Violetta N. Volkova[1], Liudmila V. Chernenkaya
Highest School of Cyberphysic Systems and Control
Institute of Computer Sciences and Technology
Peter the Great St. Petersburg State Polytechnic University
St. Petersburg, Russia
[1]violetta_volkova@list.ru

Elena N. Desyatirikova
Voronezh State Technical University
Voronezh, Russia
science2000@ya.ru

Moussa Hajali
Faculty of Informatics, Damascus University
Damascus, Syria
hajalim@hotmail.com

Almothana Khodar, Alkaadi Osama[1]
Voronezh State University
Voronezh, Russia
[1]oalkadee@gmail.com

***Abstract*—Cloud computing is the emerging internet based technology which emphasizes commercial computing. Load balancing helps in improving the performance of the centralized server. In the present work, various algorithms are analyzed using an analysis tool, namely, cloud analyst. Comparison is also made for algorithms load balancing.**

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

I. INTRODUCTION

Cloud computing is the concept of a "cloud of computing", according to which programs are launched and output the results of work in a standard web browser window on a local PC, with all applications and their data necessary for operation located on a remote server on the Internet. The advantages of cloud computing include the following: reduced requirements for the computing power of PCs, increased fault tolerance and security, the speed of data processing increases many times, costs for hardware and software, for maintenance, power are reduced, and disk space is saved.

Cloud computing is a technology that helps to exchange data and provide a lot of resources to users. Users pay only for the resources that they used. Cloud computing stores data and distributed resources in an open environment, and the amount of data storage increases very quickly. Thus, load balancing is the main task in the cloud environment. Load balancing helps to distribute the dynamic workload across multiple nodes to ensure that no node is overloaded.

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 used to distribute more load to smaller processing nodes to improve overall system performance [1]. In a cloud computing environment, load balancing needs to distribute the dynamic local workload evenly between all nodes. Load balancing helps in the fair allocation of computing resources to achieve a high level of user satisfaction and proper use of resources. High resource utilization and proper load balancing help minimize resource consumption. This helps to implement fault tolerance, scalability and avoid difficulties [2].

Load balancing is a method that has helped networks and resources, to provide maximum throughput with minimal response time. Load balancing is performed at two levels in cloud computing [3]:

- The level of the virtual machine, the mapping is made between applications that are loaded in the cloud on the virtual machine. The load balancer assigns the requested virtual machine to physical computers, which balances the load of multiple applications from the PC.
- A host level, a mapping between the virtual machine and host resources that allow processing of several incoming application requests.

III. EXISTING LOAD BALANCING POLICIES

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).

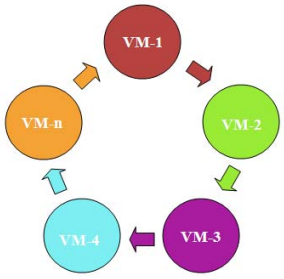

Fig. 1. Round Robin Algorithm (RR).

978-1-5386-4340-2/18/$31.00 ©2018 IEEE

Here time is divided into several sectors, and each node is given a specific time quantum or time interval, and in this quantum the node will perform its operations.

In Round Robin, scheduling a time quantum plays a very important role, because if the time slice is very large, then the Round Robin scheduling algorithm is the same as the FCFS planning [4].

The disadvantage of the method is that, although the algorithm is very simple, but to determine the quantum size, it generates an additional load on the scheduler. In addition, it has higher context switches that increase the turnaround time, and low throughput.

B. Active Load Balancing Monitoring (AMLB)

This algorithm has a dynamic character. It stores information about each VM virtual machine and the number of requests that are currently assigned to each VM. When the request is distributed by the new VM and if there are several VMs, the first recognized one is selected, and the AMLB returns the VM identifier to the data center 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).

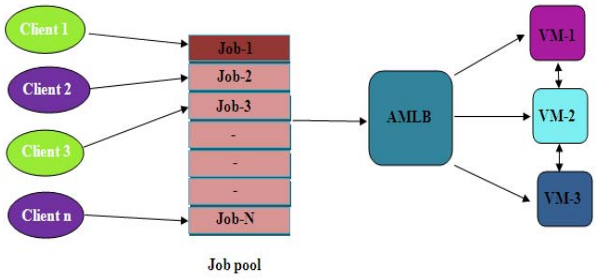

Fig. 2. AMLB Algorithm (AMLB).

The disadvantage of the algorithm is that AMLB always finds the least loaded VM to assign a new incoming request, but does not check whether it was used earlier or not (therefore some VM is used intensively, and some are still not involved).

C. Throttled Load Balancing Algorithm (TLB)

In this algorithm, the load balancer maintains a table of virtual machine indexes, as well as their states (Available or Busy). The client / server first makes a request to the data center to find a suitable virtual machine (VM) to perform the recommended task (Fig. 3).

The data center requests a load balancer to distribute the virtual machine. The load balancer scans the index table from above until the first available virtual machine is found or the index table is completely scanned.

If a virtual machine is found, the data center passes the request to the virtual machine identified by the identifier. In addition, the data center confirms the load balancing of the new distribution, and the data center appropriately revises the index 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.

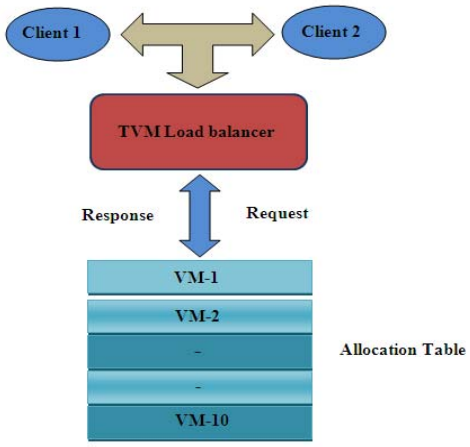

Fig. 3. Throttled Algorithm (TLB).

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.

IV. CLOUD ANALYST SIMULATOR

The simulation and analysis of the performance of the three load balancing algorithms are performed using the "Cloud Analyst" tool [5]. It allows the user to run multiple simulations with small parameter changes, and also allows you to customize the location of the users who create the application and the location of the data centers [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.

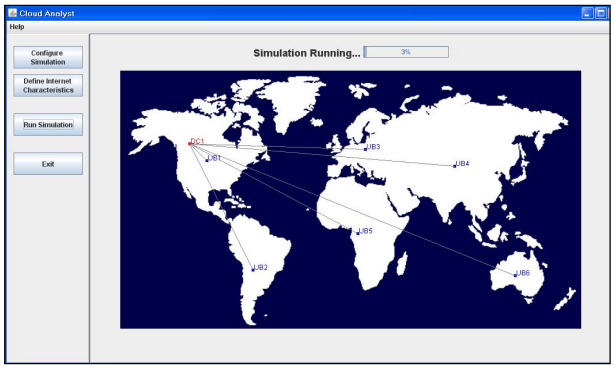

Fig. 4. Cloud Analyst Simulator.

In the modeling process, CloudSim 4.0 software was used.

V. Simulation and Experiment

Simulation and virtual experiment are the best way to test the algorithm 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 world (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	Region	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

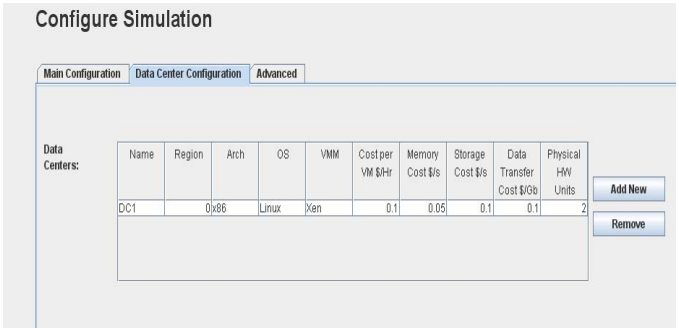

Fig. 5. Configuration of the data center.

Limit the model to the fact that each user database is contained in one time zone, and assume that most users use the application in the evenings after work about 2 hours.

Suppose also that 5% of registered users will be on-line at peak time simultaneously and only one-tenth of this number will be on the network during off-peak hours. Suppose that each user makes a new request every 5 minutes, when on-line.

VI. Simulation Results

We performed the simulation three times in accordance with the previous parameters. Each time, we changed the load balancing algorithm, which was analyzed. The results were compared 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).

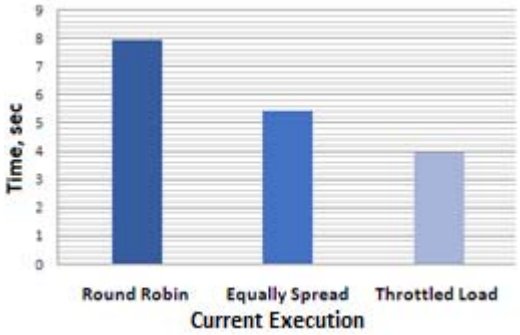

Fig. 6. Total response time.

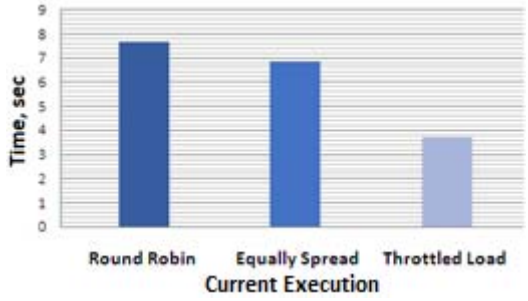

Fig. 7. Data center processing time

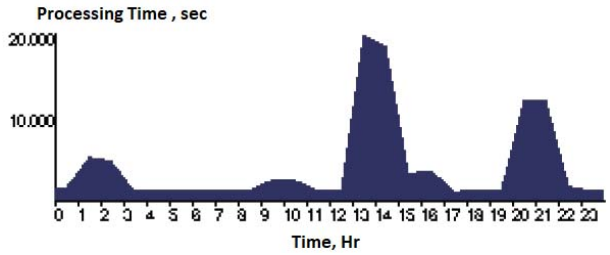

Fig. 8. Hourly data center load in (RR).

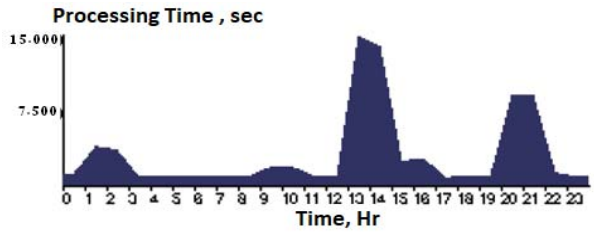

Fig. 9. Hourly data center load in (AMLB).

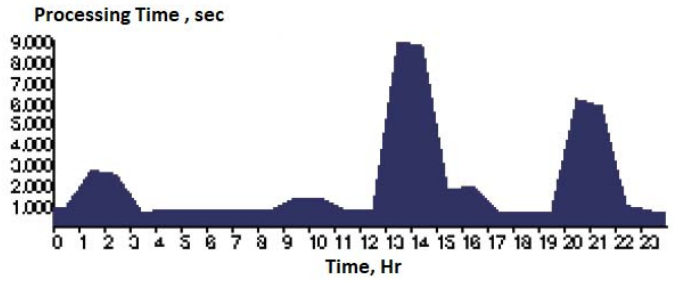

Fig. 10. Hourly data center load in (TLB).

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

REFERENCES

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 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.

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

[4] 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.

[5] 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.

[6] 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.