

Utilizing Virtual Machines Efficiently using Synchronized Throttled Algorithm in Cloud

Navyashree Shetty B

Department of Computer Science and Engineering
Canara Engineering College
Benjanapadavu, Mangalore, India

Dr. Demian Antony D'Mello

Department of Computer Science and Engineering
Canara Engineering College
Benjanapadavu, Mangalore, India

Abstract Cloud computing is network model which is used for accessing, storing and managing the data or resources present in the cloud and utilizing it for further purpose. There are various issues present in the cloud computing such as security, privacy, Service quality, Load balancing on virtual machines and soon. Load balancing is major issue in cloud environment. Load balancing is mechanism in which load is distributed among different nodes to get the best performance and to have minimum response time. Cloud computing consists of large number of network and components that are present. So, there is a need of balancing of load on machines. There are different algorithms that are implemented earlier. This paper proposes an algorithm that focuses on utilization of all virtual machine present. Experimental analysis is done using cloud analyst.

Keywords: Cloud computing, cloud analyst, virtual machine, resource utilization.

I. INTRODUCTION

In Cloud computing, cloud is a model that uses Internet where we can access, store and view the data and services on demand that can be free or at cost. Customers uses this service according to pay as you use technique. It is a network model which enables the accessing of computing resources with high speed delivery and freed with minimal effort. Load balancing is distribution of excess amount of load evenly across all nodes. Fig.1.shows the cloud computing environment. It is used to achieve user satisfaction and maximum resource utilization, making sure that no virtual machine gets over loaded. Thus, improving overall performance of the system. The load balancing consists of two categories, static and dynamic load balancing algorithms. In Static algorithm, the load is divided equally between the servers. In Dynamic, the whole network is searched and based on capacity of server the load is distributed.

The Static algorithm is easier than dynamic load balancing algorithms.

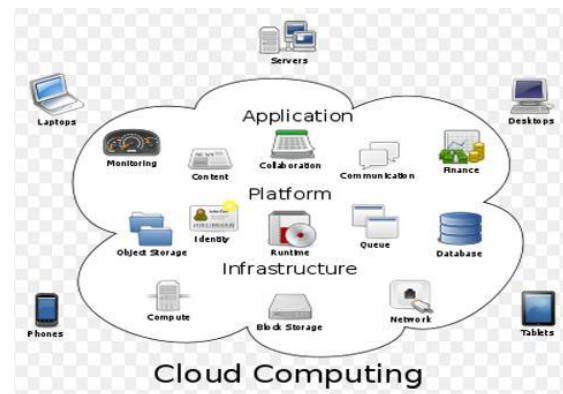


Fig.1. Cloud Computing Environment

Cloud computing includes three different service models to provide services to customers when they need a service. Software as a Service is method of providing applications and software to customers over the Internet. SaaS applications run on a SaaS provider's servers. Platform as a Service is defined as combination of database management systems, software and development tools. PaaS services are hosted by CSPs infrastructure. User can plan, develop and implement applications on provider's platform. Infrastructure as a Service refers as service model which is used to access, manage the infrastructure such as storage, networks etc. Using this service, customer can utilize resources including storage, memory space, network equipment, virtual machines and server space [1, 2]. Resource utilization is one of the performance parameter that refers the amount to which resources are utilized. Maximum utilization of resources is provided by using different load balancing algorithms. Hence, the proposed algorithm provides maximum utilization of each virtual machine that is created by data center in cloud environment. Cloud Analyst tool is used for simulation and analysis of the algorithm.

II. LITERATURE SURVEY

Load balancing is a mechanism in which load is distributed over a number of servers to decrease the traffic and to improve the performance, to minimize the response time and

to get maximum resource utilization. Survey gives the brief description of different load balancing techniques. Authors [3] proposes Round robin concept. In round robin, the load balancer will distribute the load equally between all servers based on time interval. Round robin works in circular fashion and utilizes all virtual machines present. The problem with this algorithm is that it does not check whether the virtual machine is heavily loaded or lightly loaded. It just distributes the load among all servers and does not consider the state of previous allocation of a VM for the processing the next request. Throttled Load Balancing [4] is dynamic load balancing algorithm, where it maintains allocation table of virtual machines as well as states either AVAILABLE or BUSY. Specific jobs/requests are distributed to single virtual machine that is available. If more requests are present and all allocated virtual machines are busy then the requests will be waiting until the VM becomes available.

Authors [5] propose Active Monitoring Load Balancer where it sustains the information about each VMs and to which VM the current request is allocated. This algorithm gives back the VM id to Data Center Controller (DCC), and then the request is assigned to that VM identified by that id. The proposed algorithm may lead to over/under utilization of VMs/resources. Active Monitoring Load Balancer [5] and Throttled Load Balancer [4] works well when all the virtual machines of data center have same arrangement of hardware resources and problem occurs when there exist hardware that is designed differently.

The problem with Throttled load balancing algorithm is it doesn't check whether all virtual machines are available for allocation of the load.

The problem is to develop a load balancing algorithm, which will achieve the maximum resource utilization of all virtual machines that should have different hardware design in cloud data center.

Authors in [9] proposed a novel VM- assign algorithm for allocation of request efficiently on virtual machine in cloud computing systems. The presented algorithm focus on detecting minimum loaded virtual machine and then incoming jobs are allocated on them. With the help of simulation, the proposed algorithm performs better than the Active VM-load balancer algorithm proposed in [5] and solves the problem of ineffectual utilization of VMs / resources. In [6] authors proposed Central Load Balancer (CLB) technique which solves the problem of over loading and under loading of virtual machines. CLB assigns the load to various virtual machines by comparing their priority and states. Authors explained using simulation that CLB technique performs better than Round Robin (RR) and Throttled algorithm. The problem with CLB algorithm is that it lacks in utilizing the current resource which includes processor and memory with which the distribution of load can hard and dynamic.

III. PROPOSED SYNCHRONIZED THROTTLED LOAD BALANCING ALGORITHM

The solution has been proposed for utilizing all virtual machines present in data center using Synchronized throttled algorithm. In cloud computing environment, balancing of load on virtual machines is major concern and this has to be solved using proper load balancing algorithms so that the virtual machines do not get full or get insufficient load. This algorithm is continuation of Throttled Algorithm. In this technique, the following notations as been used:

$j_1, j_2 \dots j_n$: denotes jobs

UB: Set of jobs known as user base. Single base contains n number of users.

$v_1, v_2 \dots v_n$: different virtual machines

vmid: Virtual machine which has to be checked

VM: Virtual machine which defines the state either AVAILABLE or BUSY

vmIndex: Virtual machine index which has to be checked for availability

This algorithm maintains a Hash map with vmid, vmIndex as well as states either AVAILABLE or BUSY. At first all Virtual machine will be available. Data center controller gets new request, it requests SynchronizedThrottledVmLoadBalancer to allocate job. If all VMs are allocated, then SynchronizedThrottledVmLoadBalancer starts to check the availability from first VM. If all VMs are not allocated, then checking of availability starts from next VM to the last VM.

.SynchronizedThrottledVmLoadBalancer returns the vmid to the Data center controller and update the table and change the state BUSY in vmStateList.

When VM finishes the processing of request and DCC receives the response, it will inform SynchronizedThrottledVmLoadBalancer to de allocate the VM. If more jobs are waiting, then allocation starts from next VM till the last VM. Figure 2. represent the workflow of proposed algorithm.

Algorithm: SynchronizedThrottledVmLoadBalancer Input:

Jobs j_1, j_2, \dots, j_n to be assigned to virtual machine, virtual machines v_1, v_2, \dots, v_n available at Data Center.

Output: All submitted jobs j_1, j_2, \dots, j_n are allocated one by one to each available virtual machine

- 1) vmIndex = 0
- 2) Create a Hash Map vmStateList, vmid, VirtualMachineState
- 3) DCC gets a new request
- 4) DCC checks the Synchronized ThrottledVmLoadBalancer for next job.
- 5) INCREMENT vmIndex
- 6) IF vmIndex >= vmStatesList.size() THEN

- 7) vmIndex = 0
- 8) END IF
- 9) IF vmStatesList.size() > 0 THEN
- 10) FOR each VM next to the recently allocated VM
- 11) IF VM state is AVAILABLE THEN
- 12) vmIndex = index value of checked VM
- 13) END IF
- 14) END FOR
- 15) Allocate the VM having id vmIndex
- 16) Return vmIndex to the DCC
- 17) DCC notifies Synchronized ThrottledVmLoadBalancer to update allocation table accordingly
- 18) ELSE
- 19) vmIndex = -1
- 20) Return vmIndex to the DCC
- 21) ENDF
- 22) IF VM finishes the job processing and DCC receives the response cloudlet THEN
- 23) DCC indicates Synchronized ThrottledVmLoadBalancer for VM deallocation
- 24) ENDF
- 25) IF more request/job exist THEN
- 26) Repeat from Step 3
- 27) ENDF

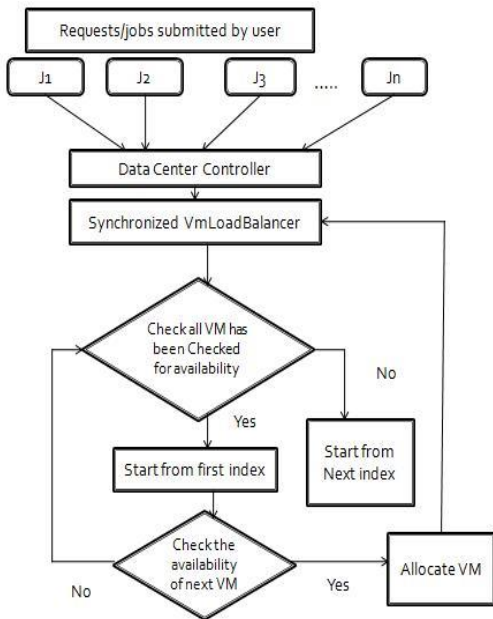


Fig. 2. Workflow of STVMLB

IV. EXPERIMENTAL SETUP IN CLOUD ANALYST TOOL

Fig. 3.shows the architecture of Cloud Analyst tool. The Cloud Analyst [7, 8] is simulation tool which expands the functionality from CloudSim toolkit. Cloud analyst is written in java and enables to configure the attributes that is required

for simulation. It is graphical user interface where it enables users to set up the following attributes and can easily execute experiments and it is repeated fashion which also benefits from highlighting the performance and accuracy of simulation thus automatically leading to overall improvement.

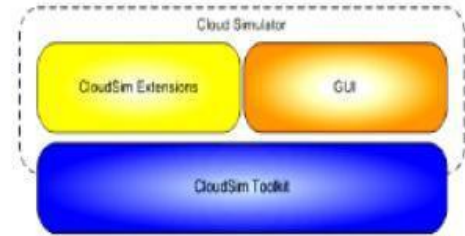


Fig.3. Architecture of Cloud Analyst

The Fig.4. shows the graphical user interface of cloud analyst where there consists a world with different regions present.



Fig. 4. Simulation Snapshot.

Fig.5. shows configuration panel where we can set up the attributes such as number of VM, number of users and data centers.

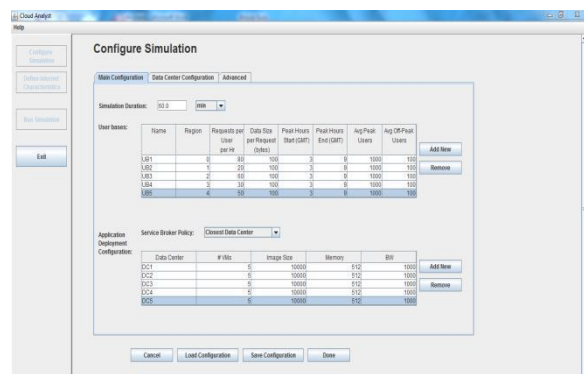


Fig. 5.Configuration Panel

According to the selection of load balancing policy, the cloud analyst gives the result to specific algorithm that is used. Fig. 6. shows selecting of load balancing algorithms.



Fig.6.Select Load balancing Policy

V. EXPECTED RESULTS

Synchronized Throttled VM Load Balancing (STVMLB) Algorithm is implemented in Java programming language and simulated using Cloud analyst tool.

Given an example, where three VM with single data center is used to configure in cloud analyst tool. Table I denote synchronized throttled load balancing algorithm gives better utilization of virtual machines rather than other algorithms. This algorithm is modified concept of Throttled load balancing algorithm; it almost uses all the virtual machines rather than Throttled algorithm. In Throttled, every time the availability of VM is checked from the first VM to allocated VM and remaining VMs will be idle or free but in STVMLB the availability of VM is checked from the first VM and later from next to last VM so that no VM will remain idle.

Table I. Response time of algorithms by using 1 DC with 3 VM

VM	Load Balancing Algorithms			
	RR	AM	Throttled	STVMLB
0	24205	33034	14852	24106
1	24205	23322	12473	24079
2	24204	20690	12285	24150

Table II. Response time of algorithms by using 1 DC with 14 VM

VM	Load Balancing Algorithms			
	RR	AM	Throttled	STVMLB
0	4842	22312	8923	4701
1	4842	5010	2967	4674
2	4842	4022	2805	4680
3	4842	3787	2763	4662
4	4842	4778	2881	4671
5	4842	5758	3165	4665
6	4842	3487	2722	4668
7	4842	4638	2815	4692
8	4842	4316	2736	4691
9	4842	4055	2688	4696
10	4842	6796	2930	4714
11	4842	3662	2544	4731
12	4842	3665	2594	4732
13	4842	4188	2507	4762

Round Robin allocates the all virtual machines and gives better result as compared to STVMLB but STVMLB checks the state of VM either it is AVAILABLE or BUSY but in Round robin, the state of VM is not checked. By considering response time, the STVMLB may give better average response time than Round Robin.

In other example, fourteen virtual machines with single data center have to configure in cloud analyst tool. After simulating, STVMLB will solve the problem of over utilization and under utilization of virtual machine.

VI. CONCLUSION

The STVMLB algorithm is designed by making modification in the concept of Throttled load balancing algorithm. Workflow diagram is used to understand the concept of STVMLB. Cloud Analyst tool is used for simulation purpose which is GUI based software. Proposed algorithm is implemented in Java language and tested in large scale cloud environment. Proposed dynamic load balancing algorithm may produce better result as to efficiently allocate the coming request and maximize the response time in cloud environment. STVMLB increases the utilization all virtual machine present in data center. Response time of STVMLB is not better than throttled algorithm and it will be improved in the future

REFERENCES

- [1] Rimal, B. P., Eunmi Choi Lumb, I., A Taxonomy and Survey of Cloud Computing Systems, Fifth International Joint Conference on INC, IMS and IDC, IEEE, E-ISBN: 978-0-7695-3769-6, Print ISBN: 978-1-4244-5209-5, pp. 44-51, 2009.
- [2] Nuaimi, K. A.; Mohamed, N.; Nuaimi, M. A. Al-Jaroodi, J., A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms, Second Symposium on Network Cloud Computing and Applications (NCCA), IEEE, Print ISBN: 978-1-4673-5581-0, pp. 137- 142, 2012.

- [3] Stuti Dave, Prashant Maheta, Utilizing Round Robin concept for load balancing algorithm at Virtual machine level in Cloud Environment, International Journal of Computer Applications, Volume 94-No 4, May 2014.
- [4] Durgesh Patel, Mr. Anand S Rajawat, Efficient Throttled Load Balancing Algorithm in Cloud Environment, International Journal of Modern Trends in Engineering and Research, pp.463-480, 2015.
- [5] Mahalle, Hemant S., Kaveri, Parag R. Chavan, Vinay, Load Balancing On Cloud Data Centres, International Journal of Advanced Research in Computer Science and Software Engineering (IJARCSSE), ISSN: 2277-128x, vol. 3, Issue. 1, pp. 1-4, 2013.
- [6] Soni, Gulshan, Kalra, Mala, A Novel Approach for Load Balancing in Cloud Data Center, Proceedings of International Advance Computing Conference (IACC), IEEE, pp. 807-812, 2014.
- [7] Wickremasinghe, Bhathiya Cloud Analyst: A CloudSim-based Tool for Modelling and Analysis of Large Scale Cloud Computing Environments, MEDC Project Report, Distributed Computing Project, CSSE Dept., University of Melbourne, pp.433-659, 2009.
- [8] Wickremasinghe, Bhathiya, Calheiros, Rodrigo N., Buyya, Rajkumar, CloudAnalyst: A CloudSim-based Visual Modeller for Analysing Cloud Computing Environments and Applications, Proceedings of 24th International Conference on Advanced Information Networking and Applications (AINA), IEEE Computer Society, pp. 446-452, 2010.
- [9] Damanal, Shridhar G. Reddy, G. Ram Mahana, Optimal Load Balancing in Cloud Computing By Efficient Utilization of Virtual Machines, Proceedings of 6th International Conference on Communication Systems and Networks (COMSNETS), IEEE, pp. 1-4, 2014.