International Journal of Technology Research and Management
ISSN (Online): 2348-9006
Vol 6 Issue 4 April 2019
NCETRCSE-2019 Conference Proceedings, CSE Dept. LNCT, Indore

An Improved Load Balancing Technique for Cloud Computing

Irshad Multani[1], Nitika kadam[2]
Computer Science, Lakshmi Narain College of Technology, Indore, 453555, India[1]
Computer Science, Lakshmi Narain College of Technology, Indore, 453555, India[2]
irshadmultani.oitsjune@gmail.com[1], kadamnitika01@gmail.com[2]

Abstract: *Load balancing is major part in cloud computing. It also play important role of cloud computing efficiency and accuracy. Many cloud load balancing algorithm are available such as Round-Robin, Throttled load balancing, Equal load share etc. These entire algorithms have some pro and cons. Good load balancing makes cloud computing more efficient and improves user satisfaction. This article introduces a better load balance model for the public cloud based on the priority and hash map concept. This paper presents a load balancing technique. This load balancing technique is based on the concept of hash map and process priority. The hash map maintains a list of available virtual machines at the data centre. The experimental results have shown that the proposed load balancing method is performing better than the existing one.*

Keywords: *Load Balancing, Public Cloud, Throttled algorithm, Cloudsim.*

1. INTRODUCTION

Cloud Computing has emerge as probably the most manageable answer for the issues which might be computation intensive. Cloud supplies the approach of sharing assets and offerings to person on demand. Virtualized useful resource and offerings can be utilized without the acquaintance of geographical variances. Cloud computing delivers mechanisms that deliberate on run time request of computing assets like storage, availability, software and so on.

The Cloud platforms distinguish among the many service style, the rate, and the nice of carrier (QoS) as good as performance. This truth brings Cloud patrons the flexibleness of freely settling on target structure from large range of Cloud platforms. However, even as this raises the difficulty of the interoperability among the one-of-a-kind Clouds [1].

Load balancing in cloud computing systems is really a challenge now. Always a distributed solution is required because it is not always practically feasible or cost efficient to maintain one or more idle services just as to fulfill the required demands. Jobs can’t be assigned to appropriate servers and clients individually for efficient load balancing as cloud is a very complex structure and components are present throughout a wide spread area. Here some uncertainty is attached while jobs are assigned.

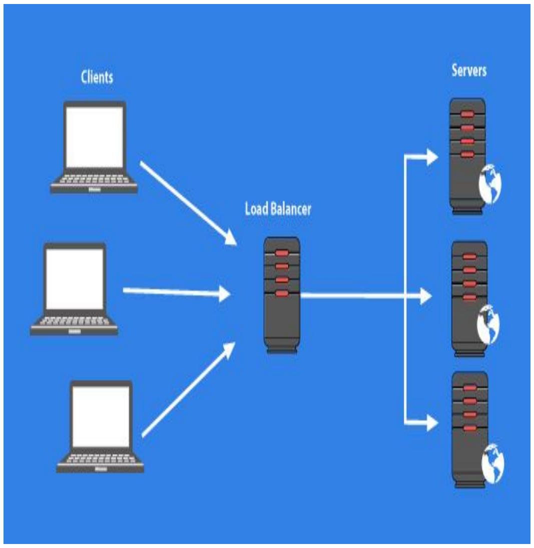

Figure 1: Concept of load balancing [2]

2. LITERATURE REVIEW

The Y. Fang et al. [3] discussed a two-level task scheduling mechanism based on load balancing to meet dynamic requirements of users and obtain high resource utilization. Algorithm achieves load balancing by first mapping tasks to virtual machines and then virtual machines to host resources thereby improving the task response time, and resource utilization also overall performance of the cloud computing environment.

Author M. Randles et al. [4] investigated a decentralized honey bee based load balancing technique that is a nature inspired algorithm for self-organization. Algorithm achieves global load balancing through local server actions. Performance of the system is enhanced with increased system diversity but throughput is not increased with an increase in system size. It is best suited for the conditions where the diverse population of service types is required.

The Z. Zhang and X. Zhang et [5] proposed a load balancing mechanism based on ant colony and complex network theory (ACCLB) in an open cloud computing federation. Proposed algorithm uses small-world and scale-free characteristics of a complex network to achieve better load balancing. Proposed technique overcomes heterogeneity is adaptive to dynamic environments and has good scalability hence helps in improving the performance of the system.

Author S. Wang et al. [6] proposed a two- phase scheduling algorithm that combines OLB (Opportunistic Load Balancing) and LBMM (Load Balance Min-Min) scheduling algorithms to utilize better executing efficiency and maintain the load balancing of the system. OLB scheduling algorithm, keeps every node in working state to achieve the goal of load balance and LBMM scheduling algorithm is utilized to minimize the execution time of each task on the node thereby minimizing the overall completion time. This combined approach hence helps in an efficient utilization of resources and enhances the work efficiency.

Author H. Mehta et al. [7] proposed a new content aware load balancing policy named as workload and client aware policy (WCAP). It uses a unique and special property (USP) to specify the unique and special property of the requests as well as computing nodes. USP helps the scheduler to decide the best suitable node for the processing the requests. This strategy is implemented in a decentralized manner with low overhead. By using the content information to narrow down the search, this technique improves the searching performance and hence overall performance of the system. It also helps in reducing the idle time of the computing nodes hence improving their utilization

Author Y. Lua et al. [8] proposed a Join-Idle-Queue load balancing algorithm for dynamically scalable web services. This algorithm provides large- scale load balancing with distributed dispatchers by, first load balancing idle processors across dispatchers for the availability of idle processors at each dispatcher and then, assigning jobs to processors to reduce average queue length at each processor. By removing the load balancing work from the critical path of request processing, it effectively reduces the system load, incurs no communication overhead at job arrivals and does not increase actual response time.

Ghosh [9] proposed another version of throttled algorithm. This version is based on the concept of user request priority. Each user request is assigned a priority. A high priority process is executed before a low priority process.

The work done by [10] proposed a novel load balancing algorithm called VectorDot. This algorithm handles the hierarchical complexity of the datacenter and multidimensionality of resource loads across servers network switches and storage in an agile data center that has integrated server and storage virtualization technologies.

The work done by [11] proposed a mechanism CARTON for cloud control that unifies the use of LB and DRL. The LB (Load Balancing) is used to equally distribute the jobs to different servers so that the associated costs can be minimized and DRL (Distributed Rate Limiting) is used to make sure that the resources are distributed in a way to keep a fair resource allocation.

[12] [1][2] addressed the problem of intra-cloud load balancing amongst physical hosts by adaptive live migration of virtual machines. The load balancing model is designed and implemented to reduce virtual machines migration time by shared storage to balance load amongst servers according to their processor or IO usage. [13] presented an event driven load balancing algorithm for real-time Massively Multiplayer Online Games (MMOG).

The work of [14] [158] present in the context of cloud computing, virtualization is used computer resources to imitate other computer resources or whole computers. This survey study has represented a wide range of overview of the research work in the field of Cloud Computing with respect of the virtualization methods.

3. PROPOSED WORK

3.1 PROBLEM STATEMENT

When a request arrive to the Datacenter Controller, it has to be allocated to one of the nodes composed the cluster, but the requests have to be distributed evenly and equally among the system to avoid workloads and degradation of system’s performance; we need another components to balance the load overall the system called Load Balancer. The Load Balancer plays a very important role in the overall response time of the cloud. In Cloud Computing Scenario Load Balancing is composed of selecting Data Center for upcoming request and Virtual machine management at individual Data Center So, how can we guarantee a good quality of service though balancing the load in the cloud? Our aim is to design a new Load Balancer to improve quality of service by optimizing load balancing in cloud computing. This load balancing technique is based on the concept of hash map and process priority. The hash map maintains a list of available virtual machines at the data centre. The experimental results have shown that the proposed load balancing method is performing better than the existing one.

3.2 Proposed Methodology:

Input:

- Data centre requests r1,r2,….., rn
- Available virtual machines vm1,vm2, ,vmn

Output:

- Data centre requests r1,r2,……,rn are allocated available virtual machines vm1,vm2,………,vmn

Process:

1. The algorithm maintains a hash map table of all the available virtual machines which their current state and the expected response time. The algorithm calculates the throughput of all virtual machines & stores it in hash map table. This state may be available or busy. At the beginning, all the virtual machines are available.
2. When data centre controller receives a request then it forwards that request to the advanced throttled load balancer. The advanced throttled algorithm sorts the list of all the available virtual machines in the descending order of their throughput. The update throttled load balancer is responsible for the virtual machine allocation. So that the job can be accomplished.
3. The algorithm scans the hash map table. It checks the status of the available virtual machine.
4. If a virtual machine with least load and the minimum response time is found.
   - Then the advanced throttled algorithm sends the VM id of that machine to the data centre controller
   - Data centre controller sends a request to that virtual machine
   - Data centre controller sends a notification of this new allocation to the updated throttled
   - The advanced throttled algorithm updates the hash map index accordingly
5. If a virtual machine with least load and the minimum response time is not found.
   - If the priority of the new request is greater than the priority of the executing request, then the executing request is switched by the new request and placed in Queue
   - If not found it queue the request
6. When the virtual machine finishes the request.
   - The data centre controller sends a notification to advanced throttled that the vm id has finished the request.
   - advanced throttled modifies the hash map table accordingly
7. If there are more requests then the data centre controller repeats step 3 for other virtual machines until the size of the hash map table is reached. Also of the size of hash map table is reached then the parsing starts with the first hash map index.

4. RESULT ANALYSIS

The proposed algorithm is implemented on Cloud Analyst. It is java based implementation tool. We have considered 2 Data Center (DC) with 20 VMs. Simulation was repeated for Round Robin, Throttled and proposed Modified Throttled algorithm respectively. With the proposed algorithm response time for request has been improved compared to other two algorithms. The following table 1 gives the information about average response time of all three algorithms.

Table 1: Result Comparison

Algorithm	Response Time (ms)
Throttled Existing	379.76
Throttled Proposed	379.64

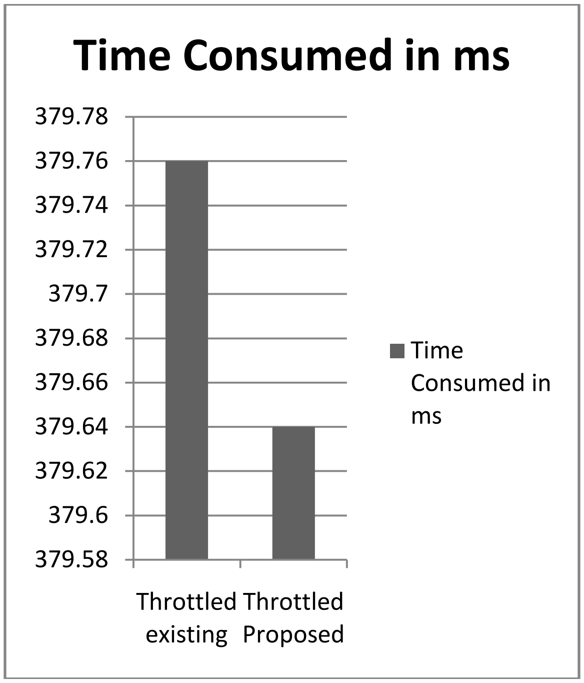

Figure 2: Response Time of all three algorithms

5. CONCLUSION

In this paper, we have proposed an updated version of the throttled load balancing algorithm. We have compared the performance of various loads balancing algorithm using the Cloud Analyst tool. The response time of proposed method is less as compared to existing load balancing techniques.

In this paper we have also proposed a survey of load balancing methods. In cloud computing load balancing is one of the main issue. When client is requesting for service it should be available to the client. When any node is overloaded with job at that time load balancer has to set that load on another free node. Therefore load balancing is necessary in cloud computing.

REFERENCES

[1] Aderem, 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, NO. 10, October 2011.

[2] David Escalante and Andrew J. Korty, “Cloud Services: Policy and Assessment”, EDUCAUSE Review, vol. 46, no. 4 (July/August 2011).

[3] Fang Y., Wang F. and Ge J.,Lecture Notes in Computer Science, 6318, 271-277,2010.

[4] Randles M., Lamb D. and Taleb-Bendiab A., 24th International Conference on Advanced Information Networking and Applications Workshops, 551-556, 2010.

[5] Infrastructure". Electronic Commerce A Managerial Perspective. pp. 27, 2008.

[6] Wang S., Yan K., Liao W. and Wang S, 3rd International Conference on Computer Science and Information Technology, 108-113, 2010.