ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

Sla Aware Load Balancing Algorithm Using Join-Idle Queue for Virtual Machines in Cloud Computing

Mehak Choudhary

M.Tech Student [CSE], Dept. of CSE, SKIET, Kurukshetra University, Haryana, India

ABSTRACT: Cloud computing is advancement of IT sector where end-user are provided with the services of infrastructure and applications on the basis of pay-per use model. Some of the cloud based application services are social networking, web hosting and content delivery. Several elements are present in cloud that is clients, datacenter and distributed servers. With cloud computing there is high availability, flexibility, less overhead for users and reduced cost. The major problem associated with cloud computing is balancing the load among cloud by choosing effective load balancing algorithm. Load in cloud can be any form of CPU load, memory capacity and delay. Load balancing is the technique to distribute the load among various nodes of distributed system for better resource utilization and response time. One very important concern is to balance the load among thousands of virtual machines. In this paper, we proposed the hybrid of two methodologies, a decentralized load balancing architecture called tldlb which provide load balancing and high availability and Join-Idle Queue(JIQ) algorithm for balancing the load among virtual machines. Two level decentralized load balancer (tldlb) uses the algorithm nn-dwrr, for dispatching large number of client requests to different virtual machine for providing services by reducing the SLA violation. But in our proposed methodology we used JIQ in place of nn-dwrr according to which initially cloudlet will be assigned to idle virtual machine and if virtual machine in virtual machine list is not idle then cloudlet will be assigned on the basis of virtual machine having minimum execution time and minimum queue length.

KEYWORDS: Cloud Computing, Load Balancing, SLA, JIQ

I. INTRODUCTION

Latest effort in delivering computing resources as a service is “Cloud Computing”. It has changed the scenario of computing from a product to be purchased to the computing here services are delivered to the consumers from large scale datacenters while accessing internet called “cloud”. Cloud computing have influenced users from hardware requirements and reducing complexities. Scalable computing and storage of resources are provided by cloud computing via internet. Infrastructure, platform and applications as services are provided by cloud computing on the basis of pay per use. Presently many companies are offering services from cloud:

- Google: It provides online Software including accessing email, text translation and Google+.
- Microsoft: Office application is provides by it in cloud which includes online storage, file sharing and hosting.
- Salesforce.com: Customers are provided with services anytime and at any location.

Cloud Computing architecture consists of many cloud computing components which are coupled loosely. Basically it is divided into two parts:

- Front End: This is basically the client side of the architecture which includes mobile devices, laptops, software, applications etc that accessed through internet. Example web browser like Firefox.
- Back End: Cloud computing is itself called “Back End”. It includes all resources, computers, services and databases that are responsible for creating the cloud.

Both ends are connected through network that is “internet”. Cloud computing uses the concept of “Server Virtualization” which allows multiple applications to run on single server rather than buying and maintaining new hardware as server. Server virtualization is the concept to divide a physical server into virtual servers for the

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

maximization of server resources. Cloud computing is categorized into two ways: 1. On the basis of services provided 2. On the basis of locality of cloud computing.
Cloud computing is divided into three types on the basis of services provided:
SaaS: Software as a service(SaaS) makes software applications available by the cloud provider. Example of SaaS are Google+, gmail email services.
PaaS: In Platform as a service(PaaS) an application development platform is provided as a service to the developer to create a web based application. Example of PaaS is Microsft Azure, Force.com.
IaaS: In Infrastructure as a service (IaaS) computing infrastructure is provided as a service to the requester in the form of Virtual Machine(VM). Example of IaaS is Amazon,Vmware..
Load Balancing is one of the major challenging issues in cloud computing. Load balancing is the technique of larger node's load distribution to smaller processing node's load distribution for improving the performance of system. To distribute dynamic workload evenly among all nodes it is required to perform load balancing in cloud computing environment. Balancing the load among virtual machines means that no available virtual machine is idle or partially loaded while others are heavily loaded.
There are many existing load balancing algorithms which are helpful in balancing the load among virtual machines and data centers.

II. RELATED WORK

Bhaskar Prasad Rimal,Eunmi Choic and Ian Lumb [1] stated the survey on cloud computing and taxonomy related to cloud computing. Cloud computing is most commonly used technique in this era which process large scale of data. For example Google processes 20 terabytes of web data. In taxonomy of cloud computing cloud architecture is discussed which have layered architecture of on demand services, these services can be accessed anywhere in the world. These services are Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS) and Hardware as a Service (HaaS). On the basis of location cloud computing is of four types private, public, hybrid and community. The proposed taxonomy will help the researcher and developer ideas of cloud computing issues for research work in future. With this paper information is provided to improve existing cloud system.

Martin Randles,David Lamb and A.Taleb Bendiab[2] stated the comparison between three distributed load balancing algorithms for cloud computing environment. Motivation for this paper was emerged from the study of optimizing the network topology with clustering. Three algorithms which are considered in this paper are firstly, for the self-organization a nature inspired algorithm that is honey-bee foraging was used. Secondly, self-organization can also be done through random sampling of system by which through local server a global load balancing is achieved which will balance all load across the system. Thirdly, for optimizing job assignment at the server system can be restructured so Active Clustering is used for this. The results of comparison were based on two phases of experiment. Initially on the basis of throughput measured versus diversity and secondly on the basis of throughput versus available resources. The result shows Active Clustering and Random Sampling Walk performs better when no of processing nodes increases and also performance variates when diversity changes.

Mohammad Alhamad,Tharam Dhillon and Elizabeth Chang [3] presents the design in cloud computing. We discussed the strategies of agreement between cloud provider and cloud consumer. This paper proposed the method for the maintenance of trust and reliability. The main contribution of the paper is to analyze the main requirement for establishing SLA model in cloud computing and explaining dynamic SLA metrics for cloud users. The paper discusses the characteristics and properties of SLA.It discusses the functional and non-functional requirements that are scalability, availability etc. SLA framework is also discussed here where two main categories of SLA metrics are discussed one is performance metrics and other is business related metrics. The metrics in IaaS for SLA are CPU capacity, boot time, scale-up etc for PaaS is integration, scalability, pay-per-use, for SaaS reliability, usability etc. For storage services geographic location, storage space, privacy is the metrics for the concern of SLA.

Yi Lua,Qiaomin Xiea,Gabriel Kliatb,Gellerb,James R.Larusb and Albert Greenberge[4] proposed an algorithm for distributed load balancing for large systems. The basic idea is to inform the dispatcher about idle processors at the time of their idleness. But here informing large number of dispatchers will increase the rate of arrival of jobs at idle processors which will increase queuing. Also informing one dispatcher will waste the cycles at idle processors and jobs assignments which further have negative impact of response time. To solve this problem, Join-Idle-Queue proposed

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

Two Level Load Balancing algorithm. For solving primary load balancing that is related to decreasing the average queue length of each processor, secondary load balancing is concerned with the availability of idle processors at each dispatcher. The JIQ has performed better in terms of response time of server.

Akshay Jain,Anagha yadav,Lohit Krishanan and Jibi Abraham[5] stated model for better load balanced environment through determining overloaded hosts and choosing best virtual machine and overloaded hosts for migration dynamically. This model includes automatic load balancing for dynamic migrations of virtual machine for maximum use of resources although manual load balancing is complex. In this model, migration decision of virtual machine is decided through the mean of CPU utilization of all hosts that is threshold value, the mean deviation of utilization of CPU of hosts around threshold value and utilization of CPU of the highest overloaded host. The proposed model uses algorithm which divided into two parts. Firstly, overloaded hosts are determined. Secondly, determination of best virtual machine for migration from overloaded hosts and destination host for migration. . This algorithm is compared with naïve migration algorithm which results in 31.75% less time to execute load than naïve algorithm for the same loading scenarios. Therefore, threshold based band algorithm is faster than naive algorithm.

Velagapudi Shreenivas,Prathap.M and Mohammad kemal [6] stated the concept of load balancing techniques in cloud computing that distributes the dynamic workload across the multiple nodes evenly so that there will be no overloading in a single node and also there will be improvement in performance and resource utilization. Load balancing is used so that the same amount of work is done by every virtual machine to increase throughput and decrease response time. In this paper some existing load balancing technologies are discussed on some parameters performance, scalability and overloading. Load balancing algorithm is classified into two categories that are static algorithm and dynamic algorithm. Static algorithm divides the traffic equally among servers. It further has Round-Robin and Weighted Round-Robin. Dynamic algorithm is that through which among whole server lightest server is selected to balance traffic.

Chung-Cheng Li and Kuochen Wang[7] proposed, tldlb (two –level decentralized load balancer) that is a load balancing architecture of decentralized approach. Scalability and high availability capability for servicing more clouds are done through tldlb because of decentralized architecture. A nn-dwrr (neural network based dynamic weighted round robin) that is a neural network based dynamic load balancing algorithm is also proposed through which dispatching of large number of requests to different virtual machines occurs, which actually provides services. The decentralized load balancer architecture has been divided into two levels that are global load balancers and local load balancers. Global load balancers are connected to SLA (Service Level Agreement) aware load balancer. Local Load Balancer has two tasks. Firstly, to monitor the load of virtual machines that are in same virtual zone concerning four metrics that CPU, memory, network bandwidth, disk input/output utilization and also response time of virtual machine. By performing experiment it was evaluated that nn-dwrr is 1.86 times faster than wrr (weighted round robin) and 1.49 times faster than capacity based and 1.21 times faster than ANN (Artificial Neural Network) based load balancing algorithm, in terms of average response time.

III. PROBLEM FORMULATED

In cloud computing architecture, a very important issue is load balancing among datacenters and at virtual machine level to maximize the throughput and service quality.

- Chung-Cheng Li [8] proposed a two-level load balancer for balancing load among virtual machines. The author proposed a neural network based dynamic weighted round-robin technique, to dispatch requests to virtual machines and reduce SLA (Service Level Agreement) violation rate by balancing load amongst virtual machines according to their load metrics. This method showed better experimental results with lesser response time.The round-robin technique has its drawback of assigning the requests to the next available virtual machine thereby creating a mismatch among SLA level and virtual machine efficiency. This is improved by using dynamic weighted round robin (dwrr) by author but the performance would have been further improved by using the dynamic weighting concept with Join- Idle Queue (JIQ) load balancing algorithm. The dynamic weighting property of the algorithm causes appropriate weighting of virtual machines and the use of Join-Idle Queue ensures the idle virtual machines with compatible weight are assigned the job first. This will enhance the performance of this architecture.

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

- Yi Lua [5] proposed Join-Idle Queue for load balancing in cloud computing system. The biggest advantage with this algorithm is that it does not incur any communication overhead between dispatchers and virtual machines at job intervals. The author associates requests to a particular virtual machine trying to maintain minimum average queue length.

IV. PROPOSED ALGORITHM

A. Description of Proposed Algorithm:

Step 1. Iterate over all virtual machines (VMs) and arrange the VM list with their MIPS.
Step 2. Calculate the threshold value on the basis of number of user requests (cloudlets) and number of virtual machines which will help in maintaining the queue length of virtual machine.
Step 3. Create SLA on the basis of response time.
Step 4. Associate cloudlet c with VMs in VM list and calculate the response time of each VM with the associated cloudlet.
Step 5. Find the compatible VM for the cloudlet with minimum response time among VM list.
Step 6. When the compatible pair of virtual machine and cloudlet is found then compare it with SLA. If pair supports SLA then add cloudlet c to VM otherwise not.
Step 7. Now when compatible pair is found it is required to check queue length of virtual machine which will be decided on the basis of threshold calculated above. Number of cloudlets in virtual machine should be less than the threshold value to maintain the minimum queue length.
Step 8. End.

V. PSEUDO CODE

Input: List of cloudlets (requests from user), List of Virtual Machines, SLA value
Output: The best solution for tasks allocation on VMs
Steps:
1. Initialize
Set Current iteration_t=1
Set Current_optimal_solution=null
2. Arrange VM according to MIPS.
3. Calculate Threshold value
Threshold value= No of requests (cloudlets) / No of VMs.
4. For c: cloudletlist
4.1. Compute compatible VMs according minimum response time
Compatible list= [];
For v: VMs
Response time of c on v
If (Response time < SLA Response time)
Add pair(c, v) to compatible list.
4.2. VM assignment
For v: VM from compatible list
If (queue length (v) < Threshold value)
Schedule[c] =v
Assign v to c
Break;
[End if]
[End for]
[End for]
5. Return

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

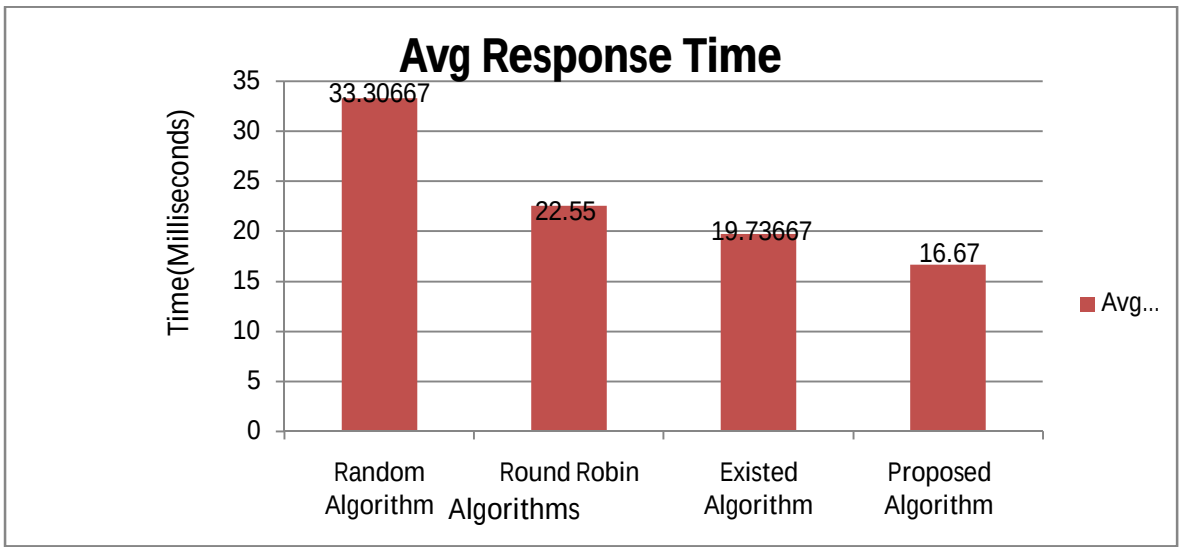
V. SIMULATION RESULTS

Simulation is performed by using cloudsim simulator. CloudSim is the toolkit for simulating cloud environment. Simulation can be defined as "running software model in any hardware".Evaluation of these strategies from different perspective from cost/profit to speed of application execution time is also done in CloudSim.
All results of simulation are performed on the basis of proposed algorithm (SLA+JIQ) and compared with existed algorithm(nn-dwrr) ,round robin ,random while providing user defined cloudlets(request for resources) and virtual machines by one user. The performance parameters which are compared are the results of average response time and average waiting time. Graphs are plotted for the performance evaluation of four different algorithms random, round-robin, existed and proposed on the basis of parameters average response time and average waiting time. Results are evaluated on the basis of 20, 30 and 40 cloudlets that is by increasing the number of cloudlets. Following table is used for the graph analysis.
Response time: Response time is the time involved in returning the results of user requests to the user.
Waiting Time: The time involved by a process in waiting in the ready queue.

Table 1: Values used for plotting graph for comparing four different algorithms by increasing number of cloudlets

Algorithms	Avg.Response Time	Avg.Waiting Time
Random	**33.30**	**20.72**
Round-Robin	**22.55**	**12.10**
Existing(nn-dwrr)	**19.73**	**9.57**
Proposed(SLA+JIQ)	**16.67**	**7.16**

1. Average response time:



Graph 7.1 Comparison of four algorithms on the basis of avg.response time while increasing cloudlets

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

Response time is the amount of time server takes to return the results of requests to the user. Response time is affected by the parameter such as network bandwidth, number of users, number and types of requests submitted.Response time is directly proportional to request being processed which means that as faster the response time, the more requests per minute are being processed.
So, average response time is given by:

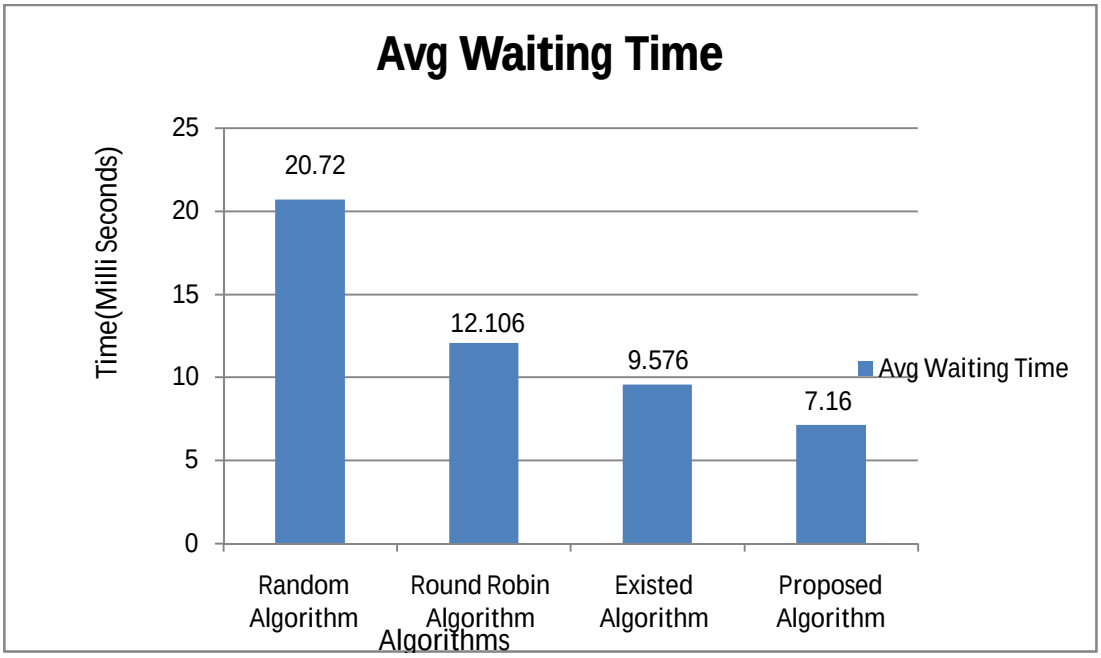
Avg response time= sum of all finishing time of cloudlets / no of cloudlets

Above graph shows the average of response time produced by four different load balancing algorithms. Average response time for the system should be minimum for the better performance. Results of graph shows that proposed method (SLA+JIQ) performs better than Round Robin, Random and existed (nn-wrr) on the basis of average response time. Here also we have used 20,30 and 40 cloudlets for the evaluation.

2. Average waiting time:
The amount time a process spent in waiting in the ready queue. Waiting time should be minimum for effective load balancing because less time will be consumed for waiting in ready queue.So, waiting time is given by:
Therefore, average waiting time is,

Waiting time= Start time – Arrival time

Avg waiting time= sum of waiting time/ no of cloudlets



Graph 7.2 Comparison of four algorithms on the basis of avg. waiting time while increasing cloudlets.

Graph drawn above shows the waiting time of four different algorithms that are random, round-robin, existed (nn-dwrr) and proposed (SLA+JIQ). For better performance and properly balancing of load there must be less waiting time in

ISSN(Online): 2320-9801
ISSN (Print) : 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 9, September 2015

ready queue. Waiting time is also analysed for the set of 20, 30 and 40 cloudlets. Results have shown that our proposed methodology (SLA+JIQ) shows less waiting time with respect to random, round-robin and existed method (nn-dwrr).

VI. CONCLUSION AND FUTURE WORK

This research presented the hybrid of two methodologies for balancing the load among virtual machines. The two methodologies which are used are SLA aware decentralized load balancing architecture (tldlb) with applying Join-idle queue algorithm in it. SLA aware decentralized load balancing architecture uses neural-network dynamic weighted round-robin(nn-dwrr) for load balancing among virtual machines but for the further improvement we have used JIQ in place of nn-dwrr. With the use of JIQ cloudlets are assigned to virtual machines on the basis of their idleness and if they are not idle they are further balanced according to the minimum response time and minimum queue length. Experimental results have shown that average response time and average waiting time for handling the particular task are reduced. In current architecture, we are focusing on the balancing of load among virtual machines only. We have discussed the cloudlet(request for resources) assignment to virtual machine without the occurrence of overloading among one virtual machine. In future, we can perform similar methodology among hosts that we can prevent overloading of host by assigning virtual machine properly among them. Also, we can perform the realistic testing of the methodology in live dataset.

REFERENCES

[1] Bhaskar Prasad Rimal ,Eunmi Choic and Ian Lumb "A Taxonomy and Survey of Cloud Computing", IEEE.pp:44-51, 2009.
[2] Martin Randles,David Lamb and A.Taleb Bendiab "A comparitive study into Distributed Load Balancing Algorithms for Cloud Computing", IEEE, pp: 551- 556, 2010.
[3] Mohammad Alhamad,Tharam Dhillon and Elizabeth Chang "Conceptual SLA Framework for Cloud Computing", IEEE, 2010.
[4] Yi Lua,Qiaomin Xiea,Gabriel Kliatb,Gellerb,James R.Larusb and Albert Greenberge "Join-Idle Queue- A novel Load Balancing Algorithm for Dynamically Scalable Web Services", ELSEVIER, 2011.
[5] Akshay Jain,Anagha yadav,Lohit Krishanan and Jibi Abraham "A Threshold based Band Model for Automatic Load Balancing in Cloud Environment", IEEE, 2013.
[6] Velagpudi Sreenivas ,Prathap.M and Mohammad kemal "Load Balancing Techniques: Major Challenge in Cloud Computing – A Systematic Review", IEEE, 2014.
[7] Chung-Cheng Li and Kuochen Wang "A SLA-aware Load Balancing Scheme for Cloud Datacenters",IEEE,pp:58-63,2014.

BIOGRAPHY

Mehak Choudhary is a research scholar pursuing M.Tech (2013-2015) in CSE, Computer Science & Engineering Department, SKIET, Kurukshetra University, Kurukshetra, Haryana, India. Her interest areas are Cloud Computing, Load Balancing and Scheduling.