IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 21, Issue 1, Ver. I (Jan - Feb 2019), PP 65-73
www.iosrjournals.org

Improve Performance by Using Load Balancing Algorithm to Reduce Response Time and Processing Time on Cloud Computing

Vijaykumar B Gadhavi[1], Prof (Dr). Kalpesh H Wandra[2]
[1]*Research Scholar ,C U Shah University*
[2] *Principal GMB Polytechnic Rajula*
Corresponding Author: Vijaykumar B Gadhavi

Abstract*: Cloud computing is one of the incredible technology which enable the new vision for IT industry. Nowadays, it has become a strong alternative for startup large as well as small scale organizations that only use the resources which actually required based on pay as per use. As Cloud Computing is growing continuously and clients from different parts of the world are demanding for the various services and better outcomes, the load balancing has become the challenge for the cloud provider. Improving response time for user requests on cloud computing is a critical issue to combat bottlenecks. As for cloud computing, bandwidth to from cloud service providers is a bottleneck. With the rapid development of the scale and number of applications, this access is often threatened by overload. Therefore, this paper our proposed Throttled Modified Algorithm(TMA) for improving the response time of Virtual machine (VM)on cloud computing to improve performance for end-user. We have simulated the proposed algorithm with the CloudAnalyts falsification tool and this algorithm has improved response times and processing time of the cloud physical or virtual infrastructure(data warehouse).*

Keywords: *Load balancing; response time; cloud computing; processing time*

Date of Submission: 12-01-2019 Date of acceptance: 27-01-2019

I. Introduction

Virtual computing is a computer model that uses ontology and process based on the Internet. The term "Green Revolution cloud" here refers to the W3c (based on its layout in the computer network topology) and dynamic level of the infrastructure contained within it. In this mathematical model, all possibilities related to information & Communication technology(ICT) are provided in the form of "services", which allow users to access services from a certain provider "in the cloud "without the knowledge and experience of that technology, nor should it consider the structure that serves that technology. Now with the explosive growth of the Internet, the exchange of data of organizations and businesses is a issue. Cloud computing Technology allows applications to be less dependent on network topology, pay per use model. All data will be uploaded to the cloud; users will only be able to access and use it anywhere, anytime. On that basis, cloud computing has increasingly developed, the problem of exchange, processing, information security and especially load balancing in cloud computing is Challenges are set for researchers as well as service providers.

Because of the new era of Advance Technology, cloud computing has become a trend, a solution in which facilities (hardware, software, networking, storage, etc) are provided promptly to the user as they request. But it is because of the explosion of data exchange that this Computing technology is posing challenges for IT People around the world, especially the load balancing on cloud based physical or virtual infrastructure(data warehouse). Because load balancing on cloud physical or virtual infrastructure(data warehouse)s is about improving response time, optimizing computing resources of the Cloud Service Provider. A few years ago, the amount of data sharing on a global network, the number of disks lined up would be two times as long as the distance to the space. It is estimated that this amount of data will increase by 44 times by 2023. The frequent use of internet data we are facing many issues related security as well as load balancing, response time execution, task completion, failure task recovery,CPU usage .One of the methods is to reduce the response time of cloud services when users request access to services. Load balancing use to find strategies to save technical resources and increase service, which directly affects the service provider's economical growth.

The research is progress in terms: 1 Introduction: basic need of load balancing in the cloud. 2 Attached task: depth look around prior work on load balancing. 3 Defined Load Balancing infrastructure. 4 Outcome of applied algorithm. 5 Outcome and appropriate Conclusion.

II. Related Work

Load balancing on cloud is subject to research around the globe & many technocrats got success in same technology.[7] Using Cloud services Load balancing play important role based on data warehouses.

Syed Madni [1] has studied the resource allocation techniques in the cloud environment. The article pointed out the Technical Criterion to improve the performance of the cloud technology. This research also outlines the base of allocating resources in the cloud, requiring resource allocation policies, strategies, and algorithms to distribute and migrate resources to best support both suppliers and users.

Shubham Sidana et al.has present NBST algorithm [2] for load balance on cloud based on the resource sort in the response of Virtual machine (VM)and will allocation of resources the request to the users. In this algorithm, authored to try to be load balancing by sort speed of Virtual machine (VM)and sort the length of the cloud. The Virtual machine (VM)list and Cloudlet(mobility-enhanced small-scale cloud datacenter) list is sent to Broker for the allocation. The list of Virtual machine (VM)and Cloudlet(mobility-enhanced small-scale cloud datacenter)s is then sent to the broker for allocation. Broker allocates through the middle point algorithm, this algorithm divides the VIRTUAL MACHINE (VM)list and Cloudlet(mobility-enhanced small-scale cloud datacenter) list until it have maximum of one Cloudlet(mobility-enhanced small-scale cloud datacenter) or one Virtual machine (VM)in the list and then allocate the resource to be executed. This algorithm allocates resources in a way that requires less processing work than allocated to high-capacity machines and vice versa. The limitation of this algorithm is that there is no mechanism for moving requests for virtual machines directly (live migration).

Feilong Tang et al. [3] and his colleagues proposed an algorithm is DLBS-dynamical load-balanced scheduling for the cloud environment proposed by. The author proposes a new method of dynamic load balancing (DLBS) to maximize throughput. Based on the development of a set of heuristic scheduling algorithms, the DLBS algorithm is effective for the OpenFlow network model, in which the data stream is balanced through time slots. Initially experimental results showed that the hypothesis was more efficient than the Round Robin(RR) and Lobus algorithms [3]. The algorithm provides the parameter $\delta(t)$ - the parameter unbalance load, through updating this parameter, the algorithm can adapt to different network conditions at all times.

Sambit Kumar [4] present a novel load balancing approach to organizing the virtualized resources of the physical or virtual infrastructure(data warehouse) efficiently. In this approach, the load to a VIRTUAL MACHINE (VM)scales up and down according to the resource capacity of the VIRTUAL MACHINE. The proposed scheme minimizes the makespan of the system, maximizes resource utilization and reduces the overall energy consumption. The proposed approach balances the load at VIRTUAL MACHINE-level to avoid overloading VIRTUAL MACHINE (VM)node. A task with a high priority gets service first to maximize the profit of Cloud Services Provider. The proposed algorithm has compared with the FCFS and RR algorithm and falsification in CloudSim(framework for modeling and falsification of cloud computing infrastructures and services) environment. The result is reduces the waiting time and optimized the makespan of the cloud physical or virtual infrastructure(data warehouse).

In this research [5], Sobhan Omranian presented a dynemic algorithm for scheduling deadline-constrained workflows is Deadline Constrained Level Based(DCLB). The algorithm used Level Load Balancing to refine deadline distribution as well as attaining lower data transfer cost in order to reach the algorithm's goals. Experimental results (based on EC2) demonstrate that DCLB compared to existing algorithms achieves higher cost efficiencies when workflow deadline is met.

Atyaf Dhari et al. [6] have proposed the LDAB scheduling algorithm to achieve load balancing and QoS. Load balancing becomes an important point to make and stabilize the system. Therefore, it is essential to improve the performance of the system by balancing the workload between virtual machines. Method: The proposed load balancing algorithm (LBDA) is to manage and balance the load between virtual machines in a physical or virtual infrastructure(data warehouse) along with reducing the completion time (Makespan) and response time. The LBDA's operational mechanism is based on three phases: first, calculate Virtual machine (VM)capacity and load on the VIRTUAL MACHINE (VM)to classify Virtual machine (VM)states (under load, load balancing, overload). Second, calculate the time needed to perform the task on each virtual machine. Finally, make the decision to distribute tasks between virtual machines based on Virtual machine (VM)state and task timing. The algorithm was compared to MaxMin, Shortest Job First and Round Robin(RR). The results of LBDA is more effective than these algorithms.

Mark van der Boor et al. [8] introduce two enhancements of the ordinary JIQ scheme where tokens are either distributed non-uniformly or occasionally exchanged among the various dispatchers. Join the-Idle-Queue (JIQ) algorithms, which rely on tokens issued by idle servers in dispatching tasks. Specifically, JIQ strategies involve minimal information exchange, and yet achieve zero blocking and wait in the many-server limit. Therefore, the author used product-form representations and fluid limits to show that the basic JIQ scheme fails to deliver zero blocking and wait for any asymmetric dispatcher loads, even for an arbitrarily low overall load.

Remarkably, it is the least-loaded dispatcher that throttles tokens and leaves idle servers stranded, thus acting as bottleneck. The enhancements of JIQ has increase in large-scale systems.

To improve the availability and continuity of cloud computing [11], the authors introduce a load balancing approach that reduces the response time and the cloud latency. By studying how to copy data from the source VIRTUAL MACHINE (VM)to the target VIRTUAL MACHINE (VM)where the source VIRTUAL MACHINE (VM)is faulty, the goal is to have users access information continuously. In paper [9], the idea is to combine the Weighted Round Robin(RR)(RR) and Max Min algorithms to form an efficient load balancing algorithm Weighted MaxMin and this algorithm has reduced two important Criterion: waiting time and response times.

Mohammad Riyaz Belgaum et al. [10], the various task scheduling algorithms are studied to present the dynamic allocation of resources under each category and the ways each of this scheduling algorithm adapts to handle the load and have high-performance computing. The task scheduling is done by the cloud service provider using preemption and non-preemption based on the requirements in a virtualized scenario which has been focused here. The results of falsification show that execution load policy is better for the physical or virtual infrastructure(data warehouse)s to use in different regions.

III. Proposed Algorithm

To improve the response time for the user (UserBase) and processing time of physical or virtual infrastructure(data warehouse). Our proposed Throttled Modified Algorithm (TMA) by effective reallocation the tasks, it had deployment at the Virtual machineLoadBalancer(virtual hardware) in Physical or virtual infrastructure(data warehouse) Controller, it was improved based on Throttled Algorithm.

A. Theoretical basis

1. Roud Robin algorithm

The Round Robin(RR) algorithm is one of the simplest algorithms based on quantum theory. Round Robin(RR) tries to distribute the load to the Virtual machine (VM)in the order of fair rotation. The idea of Round Robin(RR) is that all Virtual machine (VM)in the physical or virtual infrastructure(data warehouse) receive the same load in circular order without regard to their processing power when task allocation. This is effective for physical or virtual infrastructure(data warehouse)s that have all Virtual machine (VM)that have the same processing power. As for physical or virtual infrastructure(data warehouse)s there are large Virtual machine (VM)capable of power processing large disparities, which are ineffective.

2. Throttled Algorithm

The sequence of steps:

Step 1. Throttled Load Balancer execution load balancing by update and maintain an index table contains the status information (available '0' or not available '1') of all Virtual machine. At start, all VIRTUAL MACHINE (VM)at the status is available '0'.

Step 2. Physical or virtual infrastructure(data warehouse) Controller received a new request.

Step 3: Physical or virtual infrastructure(data warehouse) Controller query to Throttled Load Balancer for the new task.

Step 4: Throttled Load Balancer will be checked VIRTUAL MACHINE (VM)on the top table, determined the first VIRTUAL MACHINE (VM)is available.

If found VIRTUAL MACHINE:

- Throttled LoadBalancer(virtual hardware) sends the ID of VIRTUAL MACHINE (VM)to Physical or virtual infrastructure(data warehouse) Controller.
- The Physical or virtual infrastructure (data warehouse) Controller sends a request to the VIRTUAL MACHINE (VM) specified by that ID.
- Physical or virtual infrastructure(data warehouse) Controller notifies the Throttled LoadBalancer(virtual hardware) a new allocation.
- Throttled LoadBalancer(virtual hardware) updates the index and waits for new requests from the Physical or virtual infrastructure(data warehouse) Controller.

If not found VIRTUAL MACHINE

- Throttled LoadBalancer(virtual hardware) will return a value of -1 to the Physical or virtual infrastructure(data warehouse) Controller.
- The Physical or virtual infrastructure(data warehouse) Controller arranges the request.

Step 5: As for the VIRTUAL MACHINE, after processing the request and the Physical or virtual infrastructure (data warehouse) Controller receives a response, it will notify to Throttled LoadBalancer(virtual hardware) is stopped.

Step 6: If there are multiple requests, the Physical or virtual infrastructure(data warehouse) Controller repeats Step 3 with the next index and the process is repeated until the index table size is empty.

This algorithm optimizes the response time than the Round Robin(RR) algorithm[14]. But the limitation is to detect the VIRTUAL MACHINE (VM)is ready '0' with the index table size out

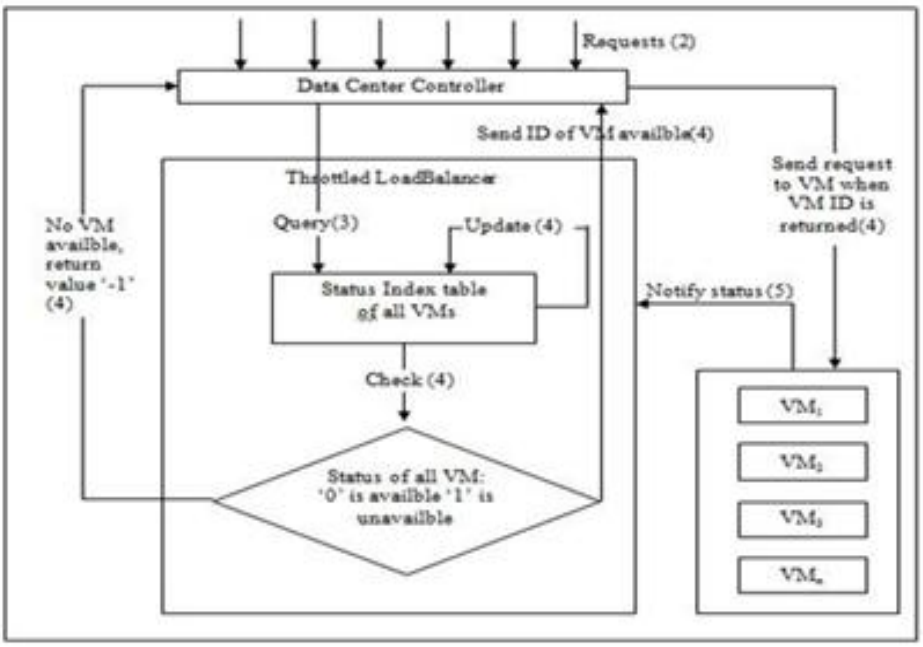

Figure 1: Throttled algorithm operation diagram

B. Proposed algorithm – Throttled Modified Algorithm (TMA)

The sequence of steps:

Step 1. The TMA Load Balancer performs load balancing by updating, maintaining two index tables.

- Available Index: Status of Virtual machine (VM)is available is '0'
- Busy Index: Status of Virtual machine (VM)is not available '1'.

At the beginning, all Virtual machine (VM)are updated in the "Available Index" table and the "Busy Index" table is empty.

Step 2. The Physical or virtual infrastructure(data warehouse) Controller receives a new request.

Step 3: Physical or virtual infrastructure(data warehouse) Controller queries to the TMA Load Balancer for next allocations.

Step 4: TMA Load Balancer detects and sends VIRTUAL MACHINE (VM)ID (VIRTUAL MACHINE) from the top down in the "Available Index" table of the Physical or virtual infrastructure(data warehouse) Controller.

- The Physical or virtual infrastructure(data warehouse) Controller sends the request to the specified VIRTUAL MACHINE (VM)by that ID.
- The Physical or virtual infrastructure(data warehouse) Controller informs the TMA Load Balancer for a new allocation.
- The TMA LoadBalancer(virtual hardware) will update this VIRTUAL MACHINE (VM)into the Busy Index and wait for the new request from Physical or virtual infrastructure(data warehouse) Controller.
- TMA Load Balancer will return a value of -1 to the Physical or virtual infrastructure(data warehouse) Controller.
- The Physical or virtual infrastructure(data warehouse) Controller arranges the request.

Step 5: As for the Virtual machine, after processing the request, and the Physical or virtual infrastructure(data warehouse) Controller receives the response from VIRTUAL MACHINE, it will notify to the TMA LoadBalancer(virtual hardware) tthen update the "Available Index" table.

Step 6: If there are multiple requests, the Physical or virtual infrastructure(data warehouse) Controller repeats Step 3 and the process is repeated until the "Available Index" table is empty.

With the our proposed algorithm (TMA), it will be possible to detect the VIRTUAL MACHINE (VM)availble (status '0') with the size of the table "Available Index" more flexible than the Throttled Algorithm. This improves the performance of the system.

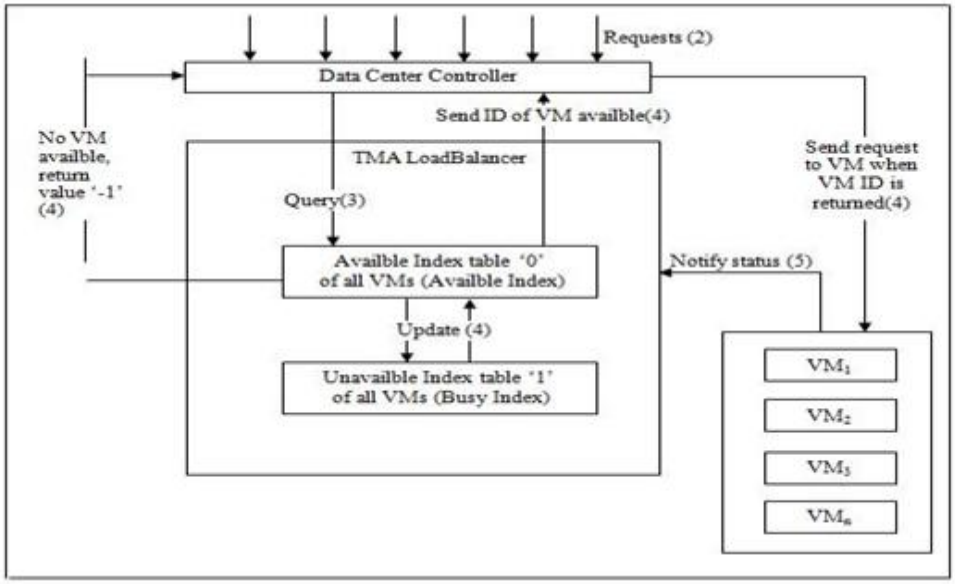

Figure 2: TMA operation diagram

IV. Falsification And Evaluation

In this paper, we used the Cloud Analyst falsification toolkit to falsification and evaluate the proposed algorithm with two algorithms: Round-Robin and Throttled. We consider the Criterion such as the overall response time of the cloud system, the processing time of the physical or virtual infrastructure (data warehouse)

C. Cloud Analyst Falsification Kit

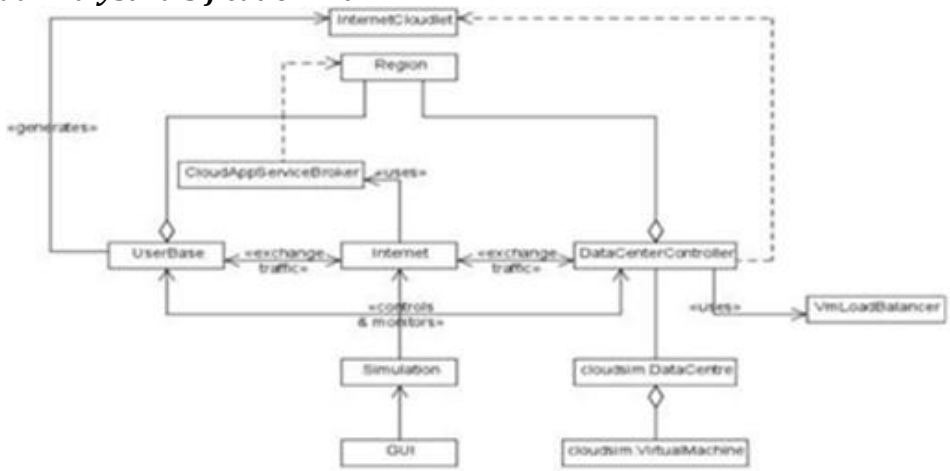

Figure 3. CloudAnalyst [7]

The implementation of the Cloud Analyst toolkit:

- UserBase will create Internet Cloudlet(mobility-enhanced small-scale cloud datacenter)s and transmit them over the Internet. Internet Cloudlet(mobility-enhanced small-scale cloud datacenter)s are assigned additional Internet Characteristics such as Latency, Transmission Delay to Physical or virtual infrastructure(data warehouse), assigned through the Cloud App Service Broker distribution policy.
- Here, the Physical or virtual infrastructure(data warehouse) Controller will decide which Virtual machine (VM)will receive or process the Internet Cloudlet(mobility-enhanced small-scale cloud datacenter) through the load balancing policies of the Virtual machine (VM)Load Balancer. And processing the Internet Cloudlet(mobility-enhanced small-scale cloud datacenter) and returning results is done under the CloudSim(framework for modeling and falsification of cloud computing infrastructures and services) background.
- After receiving the results from the VIRTUAL MACHINE (VM)returned, the Physical or virtual infrastructure(data warehouse) Controller will send back to the UserBase via the Internet and updated with Service Latency Criterion from Internet Characteristics.

- UserBase when receiving the result returned it will update the response time.
- This is repeated until the falsification is finished and the falsification results reported. Figure 3 shows the details of the falsification process.

D. Process of Falsification

We simulated 6 UserBbase that corresponds to six zones with a specific time zone, and most users use the app in the evening for about 2 hours after work. That every 5 minutes, each user sends a new request while online:

Table 1. UserBase configuration Technical Criterion.

User Base	Region	Time Zone	Peak Hour	Simulataneous Online Users During Peak Hrs	Simulataneous Online Users During Off-Peak Hrs
UB1	0	GMT - 6.00	13:00-15:00	400,000	40,000
UB2	1	GMT – 4.00	15:00 – 17:00	100,000	10,000
UB3	2	GMT + 1.00	20:00 – 22:00	300,000	30,000
UB4	3	GMT + 6.00	01:00 – 03:00	150,000	15,000
UB5	4	GMT + 2.00	21:00 – 23:00	50,000	5,000
UB6	5	GMT +10.00	09:00 – 11:00	80,000	8,000

Where:
Peak Hour: peak time period of access
Coetaneous Online Users During Peak Hrs: number of users accessing during peak times period.
Coetaneous Online Users During Off-Peak Hrs: number of users accessing during low time.
These Criterion are moulded in the Main Configuration tab of the moulded Falsification class.
Also in this tab is the configuration of the Virtual machine (VM) (Figure 4).

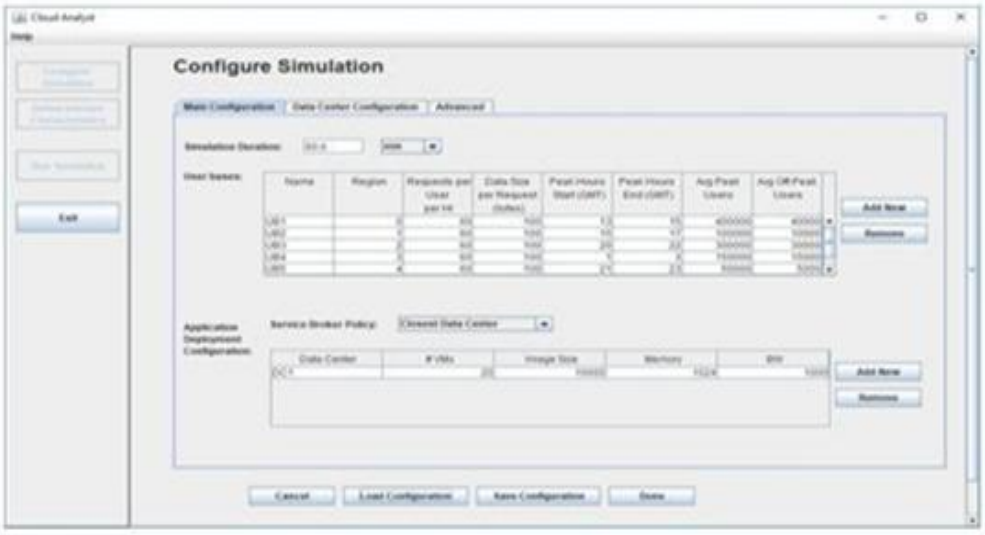

Figure 4: User and Virtual machine (VM)configuration settings

Figure 5: Physical or virtual infrastructure(data warehouse) configuration Criterion

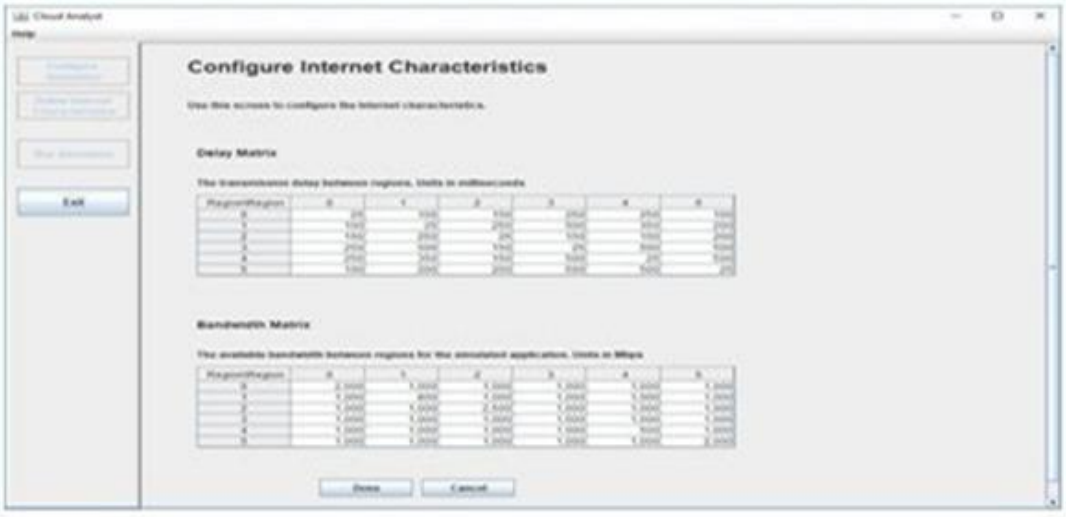

Figure 6: Internet feature configuration.

Here, simulate 3 times corresponding to 3 different policies. Specifically:
1st time: apply Round Robin(RR) policy (this policy is available in the simulator).
2nd time: apply Throttled policy
3rd time: apply the policy with our proposed TMA algorithm.

A. Falsification and Analysis Results

SCENARIO 1: Simulate with 20 virtual machines (Virtual machine).

For the Round Robin(RR) algorithm, the requests are distributed evenly over the Virtual machine (VM)so there is no need for queues to be distributed. As for the Throttled algorithm, the detection of virtual machines in the state index table by the detection method from the beginning of the table to the end of the table will lead to the status of requests to queue when the system has the number of virtual machines (VIRTUAL MACHINE) large. With the TMA, it used of two status index table (Available Index and Busy Index), the system only needs to distribute requests to Virtual machine (VM)in the Available index table without having to search for them. This eliminates the need to queue up the system, improving the processing time of the physical or virtual infrastructure(data warehouse).

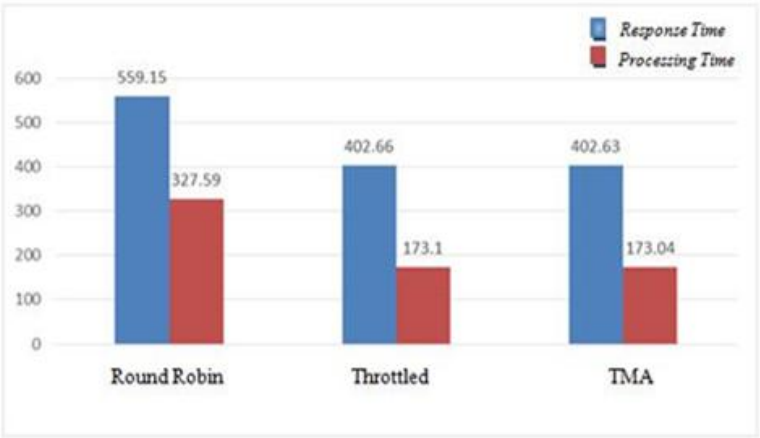

Figure 7: Falsification results with 20 Virtual machine.

In Figure 7, show that for the Round Robin(RR) algorithm, the required distribution to the VIRTUAL MACHINE (VM)rotates in a circle without considering the state of the VIRTUAL MACHINE, resulting in the Physical or virtual infrastructure(data warehouse)'s processing time and response time. The system to UserBase user base is much higher than the other two algorithms. For the other two algorithms, our TMA algorithm has a Physical or virtual infrastructure(data warehouse) processing time and the system response time is lower than the Throttled algorithm, albeit very little. Therefore, our have tried increasing the number of Virtual machine (VM)to 50 machines with the same Criterion as above for comparison again.

SCENARIO 2: Falsification with 50 virtual machines (VMs)
From Figure 8, show that the physical or virtual infrastructure(data warehouse)'s processing time and the average response time of the TMA algorithm are much lower than the Throttled algorithm when the number of Virtual machine (VM)increases.

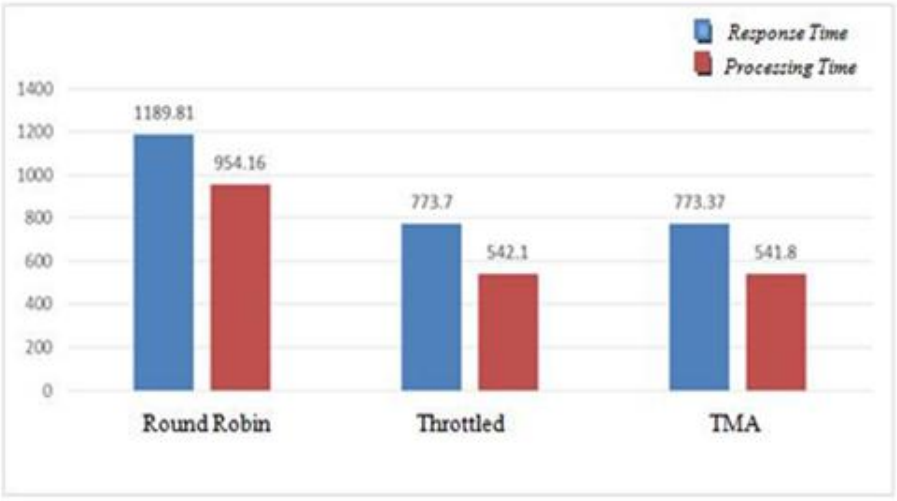

Figure 8: Falsification results with 50 Virtual machine.

From experimental results simulated in the above two cases. It helps us to see that with the TMA algorithm, the number of requests that have to queue has decreased, as well as the processing time of the physical or virtual infrastructure(data warehouse) and the response time of the system is improved than the two algorithms. This means that the TMA algorithm has better load balancing than the use of Throttled and Round Robin(RR) algorithms.

VI. Conclusion

This paper focuses on the popular load balancing algorithms in today's Cloud(Green Computing) Computing Era, analyzing and proposing an improved algorithm (TMA) based on an algorithm already in place to improve ILB(Improved Load Balancing) over older algorithms, and has accomplished the concern goals. The results obtained from the proposed algorithm have met these goals, such as limiting the number of requests queued for delivery, improving processing time and response time of hubs cloud compared to two old algorithms. This also means that with the proposed algorithm, the performance of cloud computing is improved compared to the two algorithms Round Robin(RR) and Throttled. Our proposed algorithm has shown efficiencies when the number of Virtual machine (VM)increases: reducing the response time and processing time of cloud physical or virtual infrastructure(data warehouse)s. In the future, we will study improvements to optimize the performance of the algorithm.

References

[1]. Syed Hamid Hussain Madni (2016), "Recent advancements in resource allocation techniques for cloud computing environment: a systematic review", Springer Science+Business Media New York 2016, DOI 10.1007/s10586-016-0684-4,.
[2]. Shubham Sidana, Neha Tiwari, Anurag Gupta Inall Singh Kushwaha (2016), "NBST Algorithm: A load balancing algorithm in cloud computing", International Conference on Computing, Communication and Automation (ICCCA2016), DOI: 10.1109/CCAA.2016.7813914, 29-30 April, Noida, India.
[3]. Feilong Tang, Laurence T. Yang, Can Tang, Jie Li and Minyi Guo (2016), "A Dynamical and Load-Balanced Flow Scheduling Approach for Big Physical or virtual infrastructure(data warehouse)s in Clouds", DOI 10.1109/TCC.2016.2543722,IEEE TRANSACTIONS ON CLOUD COMPUTING.
[4]. Sambit Kumar Mishra, Md Akram Khan, Bibhudatta Sahoo, Deepak Puthal, Mohammad S. Obaidat, and KF Hsiao (2017), "Time efficient dynamic threshold-based load balancing technique for Cloud Computing" IEEE International Conference on Computer, Information and Telecommunication Systems (CITS), 21-23 July, Dalian, China.
[5]. Sobhan Omranian-Khorasani and Mahmoud Naghibzadeh (2017), "Deadline Constrained Load Balancing Level Based Workflow Scheduling for Cost Optimization", 2017 2nd IEEE International Conference on Computational Intelligence and Applications, Beijing, China 8-11 Sept.
[6]. Atyaf Dhari, Khaldun I. Arif (2017), "An Efcient Load Balancing Scheme for Cloud Computing", Indian Journal of Science and Technology, Vol 10(11), DOI: 10.17485/ijst/2017/v10i11/110107.
[7]. Bhathiya Wickremasingle (2010), "Cloud Analyst: A CloudSim(framework for modeling and falsification of cloud computing infrastructures and services)- based Tool for Modelling and Analysis of Large Scale Cloud Computing Environments", MEDC Project Report, in 433-659 Distributed Computing Project, CSSE Dept, University of Melbourne.
[8]. Mark van der Boor1, Sem Borst1,2, and Johan van Leeuwaarden (2017), "Load Balancing in Large-Scale Systems with Multiple Dispatchers", IEEE INFOCOM 2017 - IEEE Conference on Computer Communications, 1-4 May, Atlanta, GA, USA.

[9]. Bharat Khatavkar, Prabadevi Boopathy (2017), "Efficient WMaxMin Static Algorithm For Load Balancing In Cloud Computation", DOI: 10.1109/IPACT.2017.8245166, International Conference on Innovations in Power and Advanced Computing Technologies [i-PACT2017], Vellore, India.
[10]. Guo Xiao, Wu Wenjun, Zhao Jiaming, Fang Chao and Zhang Yanhua (2017), " An OpenFlow Based Dynamic Traffic Scheduling Strategy for Load Balancing," 2017 3rd IEEE International Conference on Computer and Communications (ICCC),DOI: 10.1109/CompComm.2017.8322602, 13-16 Dec, Chengdu, China.
[11]. Jananta Permata Putra, Supeno Mardi Susiki Nugroho, Istas Pratomo (2017), "Live Migration Based on Cloud Computing to Increase Load Balancing", 2017 International Seminar on Intelligent Technology and Its Application, 10.1109/ISITIA.2017.8124096, Publisher: IEEE, 28-29 Aug. Surabaya, Indonesia.
[12]. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems, 25:599616, 2009.
[13]. Kansal Nidhi Jain, "Cloud Load Balancing Techniques: A Step Towards Green Computing", IJCSI International Journal Of Computer Science Issues, January 2012, Vol. 9, Issue 1, No 1, Pg No.:238-246, ISSN (Online): 1694-0814.
[14]. Rimal B.P., Choi E. and Lumb I. (2009) 5th International Joint Conference on INC, IMS and IDC, 44-51.
[15]. Bhathiya, Wickremasinghe.(2010)"Cloud Analyst: A Cloud Sim-based Visual Modeller for Analysing Cloud Computing Environments and Applications".
[16]. Zenon Chaczko, Venkatesh Mahadevan, Shahrzad Aslanzadeh, Christopher Mcdermid (2011)"Availabity and Load Balancing in Cloud Computing" International Conference on Computer and Software Modeling IPCSIT vol.14 IACSIT Press, Singapore 2011.
[17]. Roderigo N. Calherios, Bhathiya Wickremasinghe "Cloud Analyst: A Cloud-Sim-Based Visual Modeler For Analyzing Cloud Computing Environments And Applications". Proc of IEEE International Conference on Advance Information Networking and Applications, 2010.
[18]. Chaczko Zenon, Venkatesh Mahadevan, Shahrzad Aslanzadeh, Christopher Mcdermid (2011)"Availabity and Load Balancing in Cloud Computing" International Conference on Computer and Software Modeling IPCSIT vol.14 IACSIT Press, Singapore 2011.
[19]. Sharma Sandeep, Sarabjit Singh, and Meenakshi Sharma "Performance Analysis of Load Balancing Algorithms" World Academy of Science, Engineering and Technology 38 ,2008 page no 269- 272.
[20]. Hiranwal Saroj , Dr. K.C. Roy, "Adaptive Round Robin(RR) Scheduling Using Shortest Burst Approach Based On Smart Time Slice" International Journal Of Computer Science And Communication July-December 2011 ,Vol. 2, No. 2 , Pp. 319-323.

Author Profile:

Vijaykumar B Gadhavi is a Assistant Professor in the Computer Engineering Department at the Shree Swaminarayan institute of Technology & Research Scholar of Computer Engineering Department ,C U Shah University ,Surendranagar,Gujarat . He has taught a number of Subjects of Computer Engineering over the years, as well as more general courses on Cloud Computing, Cyber security and other Advance Technolgoies. His research and publication interests include Load Balancing,Data Security (Forensic Science and E Governance) and AI.

Dr. K. H. Wandra Birth date 10th September, 1974, Surendranagar (Gujarat), India. Ph.D. from Saurashtra University, Rajkot, On "Role of Wireless Technology for Mobile Augmented Reality System(MARS)", done M.E. Microprocessor System Applications from M.S. University, Vadodara. in year 1999 & B.E. in Electronics & Communication form North Gujarat University, Patan, in year1995. having more than 16 years of UG & 7 years of PG(MCA and M.E.) Teaching Experience, Written more than 10 Books related to Computer & IT related area e.g (1) Java Programming Advanced Topics by CENGAGE Learning Publication, (2) Mobile Computing (3) Introduction Of Computers (4) Computer Graphics by Akshat Publication , Gujarat. Published / presented Five International and 7 national Research Papers. Guided more than 10 PG Students and more than 155 UG Students Dissertation work. Dr. K. H. Wandra is a Chairman of Board of Studies Computer Engineering At Saurashtra University, Rajkot, Core Committee member For Syllabus of UG & PG at Gujarat Technology University, Ahmedabad. Section Managing Committee Member of ISTE Gujarat Section, Life member of ISTE, member of IEEE and CSI, Interested for working in the area of Wireless communication ,Networks and Advanced Microprocessors.