

EXECUTION ANALYSIS OF LOAD BALANCING ALGORITHMS IN CLOUD COMPUTING ENVIRONMENT

Soumya Ray and Ajanta De Sarkar

Department of Computer Science and Engineering, Birla Institute of Technology, Mesra,
Kolkata Campus, Kolkata 700 107
soumyaray@bitmesra.ac.in
adsarkar@bitmesra.ac.in

ABSTRACT

The concept of Cloud computing has significantly changed the field of parallel and distributed computing systems today. Cloud computing enables a wide range of users to access distributed, scalable, virtualized hardware and/or software infrastructure over the Internet. Load balancing is a methodology to distribute workload across multiple computers, or other resources over the network links to achieve optimal resource utilization, maximize throughput, minimum response time, and avoid overload. With recent advent of technology, resource control or load balancing in cloud computing is main challenging issue. A few existing scheduling algorithms can maintain load balancing and provide better strategies through efficient job scheduling and resource allocation techniques as well. In order to gain maximum profits with optimized load balancing algorithms, it is necessary to utilize resources efficiently. This paper presents a review of a few load balancing algorithms or technique in cloud computing. The objective of this paper is to identify qualitative components for simulation in cloud environment and then based on these components, execution analysis of load balancing algorithms are also presented.

KEYWORDS

Cloud computing, load balancing, simulation, CloudSim.

1. INTRODUCTION

Cloud computing is a distributed computing paradigm that focuses on providing a wide range of users with distributed access to scalable, virtualized hardware and/or software infrastructure over the internet [17]. Potentially it can make the new idea of ‘computing as a utility’ in the near future. Despite this technical definition cloud computing is in essence an economic model for a different way to acquire and manage IT resources. An organization needs to weigh cost, benefits and risks of cloud computing in determining whether to adopt it as an IT strategy. The availability of advance processors and communication technology has resulted the use of interconnected, multiple hosts instead of single high-speed processor which incurs cloud computing.

Recently, *public cloud* is made available as a pay per usage model while *private cloud* can be built with the infrastructure of the organization itself. Web Services, Google AppEngine, and Microsoft Azure are examples of *public cloud*. The service provided by the *public cloud* is known as utility computing. As benefit, users can access this service “anytime, anywhere”, share data

and collaborate more easily, and keep data safely in the infrastructure. Although there are risks involved with releasing data onto third party servers without having the full control of it.

In cloud computing environment, the random arrival of tasks with random utilization of CPU service time requirements can load a specific resources heavily, while the other resources are idle or are less loaded [11]. Hence, resource control or load balancing is major challenging issue in cloud computing. Load balancing is a methodology to distribute workload across multiple computers, or other resources over the network links to achieve optimal resource utilization, maximize throughput, minimum response time, and avoid overload. This research work is based on simulation technique and it uses the cloud simulator, CloudSim [7].

The objective of this paper is not only to study a few load balancing algorithms or techniques in cloud computing, it also presents execution analysis of the algorithms with various combination of components in CloudSim. Components of the clouds and different types of cloud systems are discussed in Section 2. Section 3 presents benefits or advantages and significant shortcomings of cloud computing. Concepts of load balancing in cloud computing and discussion of a few algorithms are explained in Section 4. Section 5 explains the importance of simulation technique in cloud environment. Proposed execution environment is presented in Section 6. Section 7 represents experimental results of executing task in CloudSim. Section 8 presents related work and Section 9 concludes the paper with directives of future work.

2. COMPONENTS AND TYPES OF CLOUD SYSTEM

Cloud computing is the dynamic provisioning of information technology capabilities (hardware, software, or services) from third parties over a network [1]. Any cloud computing system consists of three major components such as clients, datacenter and distributed servers. These components are shown in Figure 1. Brief discussion of specific role and purpose of each component is presented in the following:

- a) *Client*: End users interact with the clouds to manage information related to the cloud. Clients generally fall into three categories-
 - Mobile: windows mobile smart phone like a blackberry or an I Phone.
 - Thin: They don't do any computation work. They only display the information. Servers do all the work for them. The clients don't have any internal memory.
 - Thick: These use different browsers like *internet explorer* or *mozilla firefox* or *google chrome* to connect to the different cloud.
- b) *Datacenter*:: Datacenter is nothing but collection of servers hosting different applications. An end user connects to the datacenter to subscribe different applications. A datacenter may exist at a large distance from the clients.

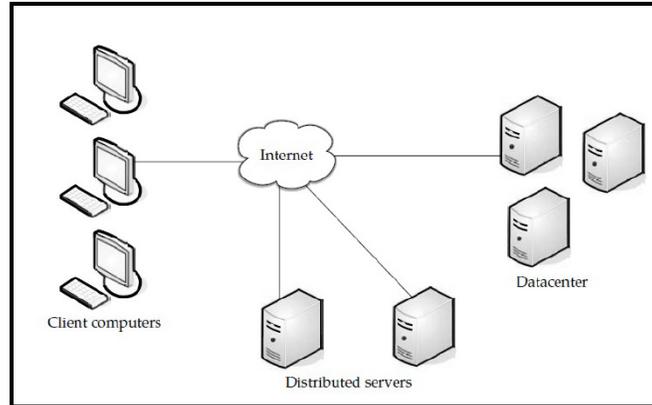


Figure 1: Components of Cloud [12]

- c) *Distributed Servers*:: A server, which actively checks the services of their hosts, known as Distributed server. Distributed servers are the part of a cloud which is available throughout the internet hosting different applications. But while using the application from the cloud, the user would feel that he /she is using this application from its own machine.

The types of cloud computing technology can be viewed from two perspectives: *Capability* and *Accessibility* [5]

- a) *Based on Type of Capability*: According to this categorizations, cloud system provides three different types of services as follows:
- *Software-as-a-Service (SaaS)*: *SaaS* focuses on providing users with business specific capabilities such as email or customer management. In *SaaS* organizations and developers can use the business specific capabilities developed by third parties in the “cloud”. One of the example of *SaaS* provider is *Google Apps* that provides large suite of web based applications for enterprise use.
 - *Platform-as-a-Service (PaaS)*: *PaaS* is a service model of cloud computing. In this model clients create the software using tools and libraries from the provider. Clients also control software deployment and configuration settings. The provider provides the network, servers and storage. One of the examples of *PaaS* is *Google App Engine* that provides clients to run their applications on Google’s infrastructure.
 - *Infrastructure-as-a-Service (IaaS)*: *IaaS* provides mainly conceptual infrastructure over the internet (e.g. compute cycles or storage). *IaaS* allows organizations and developers to extend their IT infrastructure on demand basis. One of the examples of *IaaS* providers is *Amazon Elastic Compute Cloud (EC2)*. It provides users with a special virtual machine that can be deployed and run on the *EC2* infrastructure.
- b) *Based on Accessibility Type*: On the basis of accessibility clouds are of two types, as mentioned in the following:
- *Public Cloud*: In public cloud resources are offered as a service, over an internet connection, for a pay-per-usage fee. Clients do not need to purchase hardware to get the service and can also scale their use on demand.
 - *Private Cloud*: In private clouds resources are deployed inside a firewall and managed by the client’s organization. Organization owns the hardware and software infrastructure, manages the cloud and controls access to its resources.

3. BENEFITS AND BARRIERS OF CLOUD COMPUTING

Progress of Cloud Computing is enormous with respect to personal uses and business uses. Users of cloud computing can utilize or maintain the online resources. Among several advantages or benefits, few are discussed below:

- a)* Scalability: Scalability is the capability of a system to increase total throughput under an increased load when resources are added. Resources can be hardware, servers, storage, and network. The user can quickly scale up or scale down the resources in cloud computing according to their need without buying the resources.
- b)* Virtualization: In cloud computing, virtualization is a concept where users have a single view of available resources irrespective of their arrangement in physical devices. So it is advantageous for the providing the service towards users with less number of physical resources.
- c)* Mobility: Cloud Computing means mobility because users can access applications through internet easily at any point of time.
- d)* Low Infrastructure Costs: The pay-per-usage model is supported in cloud computing. It actually helps an organization to pay for the resources they need, not to make any investment for the resources available in the cloud. Moreover, the provider does not require any infrastructural maintenance or upgrade costs.
- e)* Increased Storage: Users or clients in cloud computing can store more data in cloud than on private computer systems, which they use regular basis. It not only relieves them from buying extra storage space, but also improves performance of their regular system, as it is less loaded. On the other hand, data or programs are accessed anytime through internet, since they are available in cloud.

Thus, the cloud computing provides several advantages with form of elasticity, availability and expandability on-demand. Still it has some constraints or limitations as discussed in the following:

- a)* Latency: Low latency has always been an important consideration in telecom networks for voice, video and data. As cloud-based architecture can easily be accessed through the internet, so high latency is an important issue in every communication between client and provider.
- b)* Platform or Language constraints: Adaptation of platform or language always plays an important role. Till today, cloud providers support specific language or platform that does not interoperable with other providers. So a universal set of standards needs to be defined in case of language or platform adaptation.
- c)* Resource Control: Controlling of resources in cloud is not always in scope of client. It may vary between different cloud providers. Sometimes resource isolation is very much needed but it is very hard to isolate for the client to identify the exact resource. At the same time, resources may exhaust for keeping data or providing services, so data or program may need to migrate over other resources. This is also major and challenging issues in cloud computing. So, controlling resources and distributing loads through migration (if possible) between different resources is very much essential.

4. LOAD BALANCING IN CLOUD COMPUTING

Load balancing is the process of distributing the load among various resources in any system. Thus load need to be distributed over the resources in cloud-based architecture, so that each resources does approximately the equal amount of task at any point of time. Basic need is to provide some techniques to balance requests to provide the solution of the application faster.

Cloud vendors are based on automatic load balancing services, which allow clients to increase the number of CPUs or memories for their resources to scale with increased demands. This service is optional and depends on the clients business needs. So load balancing serves two important needs, primarily to promote availability of Cloud resources and secondarily to promote performance [24].

In order to balance the requests of the resources it is important to recognize a few major goals of load balancing algorithms:

- a) Cost effectiveness: primary aim is to achieve an overall improvement in system performance at a reasonable cost.
- b) Scalability and flexibility: the distributed system in which the algorithm is implemented may change in size or topology. So the algorithm must be scalable and flexible enough to allow such changes to be handled easily.
- c) Priority: prioritization of the resources or jobs need to be done on before hand through the algorithm itself for better service to the important or high prioritized jobs in spite of equal service provision for all the jobs regardless of their origin.

Brief reviews of few existing load balancing algorithms are presented in the following:

- I. *Token Routing:*** The main objective of the algorithm [24] is to minimize the system cost by moving the tokens around the system. But in a scalable cloud system agents can not have the enough information of distributing the work load due to communication bottleneck. So the workload distribution among the agents is not fixed. The drawback of the token routing algorithm can be removed with the help of heuristic approach of token based load balancing. This algorithm provides the fast and efficient routing decision. In this algorithm agent does not need to have an idea of the complete knowledge of their global state and neighbor's working load. To make their decision where to pass the token they actually build their own knowledge base. This knowledge base is actually derived from the previously received tokens. So in this approach no communication overhead is generated.
- II. *Round Robin:*** In this algorithm [25], the processes are divided between all processors. Each process is assigned to the processor in a round robin order. The process allocation order is maintained locally independent of the allocations from remote processors. Though the work load distributions between processors are equal but the job processing time for different processes are not same. So at any point of time some nodes may be heavily loaded and others remain idle. This algorithm is mostly used in web servers where Http requests are of similar nature and distributed equally.
- III. *Randomized:*** Randomized algorithm is of type static in nature. In this algorithm [25] a process can be handled by a particular node n with a probability p . The process allocation order is maintained for each processor independent of allocation from remote processor. This algorithm works well in case of processes are of equal loaded. However, problem arises when loads are of different computational complexities. Randomized algorithm does not maintain deterministic approach. It works well when Round Robin algorithm generates overhead for process queue.
- IV. *Central queuing:*** This algorithm [13] works on the principal of dynamic distribution. Each new activity arriving at the queue manager is inserted into the queue. When request for an activity is received by the queue manager it removes the first activity from the queue and sends it to the requester. If no ready activity is present in the queue the request is buffered, until a new activity is available. But in case new activity comes to the queue while there are unanswered requests in the queue the first such request is removed from the queue and new activity is assigned to it. When a processor load falls under the threshold then the local load manager sends a request for the new activity to the central load manager. The

central manager then answers the request if ready activity is found otherwise queues the request until new activity arrives.

- V. *Connection mechanism*: Load balancing algorithm [19] can also be based on least connection mechanism which is a part of dynamic scheduling algorithm. It needs to count the number of connections for each server dynamically to estimate the load. The load balancer records the connection number of each server. The number of connection increases when a new connection is dispatched to it, and decreases the number when connection finishes or timeout happens.

Table 1 presents a comparative study of the above-mentioned load balancing algorithms:

Algorithm	Nature	Environment	Process migration	Resource Utilization	Steadiness
Token Routing	dynamic	decentralized	possible	more	unstable
Round Robin	static	decentralized	not possible	less	stable
Randomized	static	decentralized	not possible	less	stable
Central Queuing	dynamic	centralized	not possible	less	unstable
Least Connection	dynamic	centralized	not possible	less	stable

Table 1: Comparative Study of Load Balancing Algorithms

Thus randomness of load arrival from various end cause non uniform loads on the server. After studying various load balancing algorithms, it is necessary to execute at least two or three algorithms to observe where they are improving and lacking as well. In real life scenarios, executions of such algorithms are not possible in exact cloud computing environment. As a consequence, simulating of these algorithms might be supportive in order to accomplish this research.

Next section elaborates simulation in cloud environment along with importance of simulation and brief explanation of CloudSim architecture.

5. SIMULATION IN CLOUD: CLOUDSIM

Resources and software are shared on the basis of client's demand in cloud environment. Essentially, dynamic utilization of resources is achieved under different conditions with various previous established policies. Sometime it is very much difficult and time consuming to measure performance of the applications in real cloud environment. In this consequence, simulation is very much helpful to allow users or developers with practical feedback in spite of having real environment.

This section portrays the importance of simulation technique and simulation in cloud. In this research work, simulation is carried out with a specific cloud simulator, CloudSim [7].

5.1 Importance of simulation technique:

Simulation means mimicking the actual environment towards benefit of the research. The user or researcher can actually analyze the proposed design or existing algorithms through simulation. They can check the efficiency and merit of the design before the actual system is constructed. Simulation is advantageous to the users, as they can explore the benefit of that design repeatedly. This actually reduces the cost of reconstruct as changes have been made during design time. Simulation technique provides lots of advantages as the experiments can be carried out with

voluminous data in different abstraction level. Simulators easily make available various kind of virtual environment for verification and performance evaluation of the experimented system. Even most of the time researchers could carry out benchmark experiments repeatedly in scalable environment for evaluating different aspects.

5.2 Cloud Simulator- CloudSim

Users are capable of accessing shared resources through utilizing available public cloud platform. However, accessing real cloud environment or public cloud is not always handy. Instead of the real environment, cloud simulator could facilitate the experiments.

Simulation environment allows customers or users to tune the performance bottlenecks or evaluates different kinds of features under varying load distributions[7]. Different kinds of functionalities of CloudSim are presented in the following [4].

- support for modeling and simulation of large scale cloud computing data centers
- support for modeling and simulation of virtualized server hosts, with customizable policies for provisioning host resources to virtual machines
- support for modeling and simulation of energy-aware computational resources
- support for modeling and simulation of datacenter network topologies and message-passing applications
- support for modeling and simulation of federated clouds
- support for dynamic insertion of simulation elements, stop and resume of simulation
- support for user-defined policies for allocation of hosts to virtual machines and policies for allocation of host resources to virtual machines

Besides these above-mentioned functionalities, while developers or researchers, uses CloudSim features, need not to think about the lower level details of cloud based infrastructure and services. The architecture of CloudSim comprises of four layers, as shown in

Figure 2. At the bottom most layer, *cloud resources* (hosts, datacenters) are managed. During the simulation period, these core entities are instantiated and executed. On top of this layer, *cloud services*, like allocation of CPU, memory, storage and bandwidth are provided as dedicated management interfaces. Another two top most layers are *virtual machine services* and *user interface structures*. Virtual machine in *user interface structures* layer is responsible for physical host specifications such as number of machines and their configurations. CloudSim facilitates a specific host to be concurrently shared among different VMs based on user-defined QoS specifications [4].

Next section presents proposed execution environment in order to analyse execution performance of existing load balancing algorithms.

6. PROPOSED EXECUTION ENVIRONMENT

Equal load distribution may improve resource utilization by transferring load from heavily loaded server to the idle server. Existing scheduling algorithms [13], [19], [24],[25] estimate system parameters such as the job arrival rate, CPU processing rate, and load on the processor for migrating jobs into least loaded processors in order to balance load. This research work considers Datacenter, Virtual Machine (VM), host and Cloudlet components from CloudSim for execution analysis of a few algorithms. Datacenter component is used for handling service requests. VM consist of application elements which are connected with these requests, so Datacenter's host

components should allocate VM process sharing. In [2], Bhatiya presents that Cloud Analyst can evaluate any algorithm or application deploying in the cloud. VM life cycle starts from provisioning of a host to a VM, VM creation, VM destruction, and VM migration[2].

Figure 3 represents a class diagram of the CloudSim components considered for execution analysis.

In addition to that, a brief description of these components and the working relationship between them is presented in the following:

Datacenter: Datacenter encompasses a number of hosts in homogeneous or heterogeneous configurations (memory, cores, capacity, and storage). It also creates the bandwidth, memory, and storage devices allocation.

Virtual Machine (VM): VM characteristics comprise of memory, processor, storage, and VM scheduling policy. Multiple VM can run on single hosts simultaneously and maintain processor sharing policies.

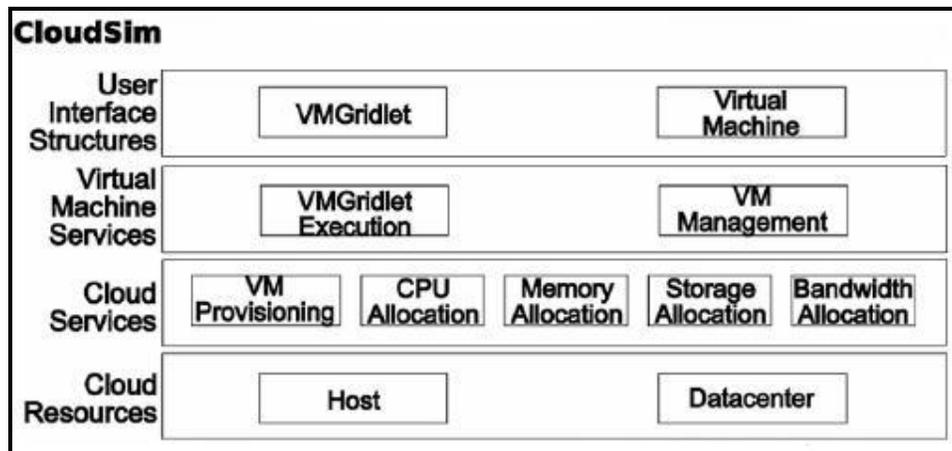


Figure 2 : CloudSim Architecture [2]

Host: This experiment considers VM need to handle a number of cores to be processed and host should have resource allocation policy to distribute them in these VMs. So host can arrange sufficient memory and bandwidth to the process elements to exeute them inside VM. Host is also responsible for creation and destruction of VMs.

Cloudlet: Cloudlet is an application component which is responsible to deliver the data in the cloud service model. So the length, and output file sizes parameter of Cloudlet should be greater than or equal to 1. It also contains various ids for data transfer and application hosting policy.

Experimental results of executing task in CloudSim are represented in the next section.

7. EXECUTION OF TASKS IN CLOUDSIM

The above-mentioned proposed execution setup is used as simulation setup for experiment purpose. This research use CloudSim-3.0 as a framework in the simulator environment. Implementation has been started with installation of simulation package CloudSim-3.0 [7] on

Windows XP (Service pack 3). Thereafter Java version 7 [10] is installed and classpath along with other necessary execution setup requirement is fulfilled.

The minimum requirement of this experiment is VM memory of 1GB, VM bandwidth of 1000 and local operating system used as a host. In this simulation setup, three well-known load balancing algorithms ‘round robin’, ‘central queuing’ and ‘randomized’ have been executed with various combinations of millions instructions per second (MIPS) vs. VM and MIPS vs. HOST. Analysis is being carried out with respect to the response time as output. Figure 4, Figure 5 and Figure 6 represent various response time based with different combinations of MIPS vs. VM and MIPS vs. Host. It is observed for all the cases, response time is inversely proportionate with MIPS vs. VM and MIPS vs. Host. But optimum response time is achieved with same value of MIPS vs. VM and MIPS vs. Host. This execution analysis illustrates that nature of each simulation results are similar as this research currently concentrates on improvement of response time with similar setup.

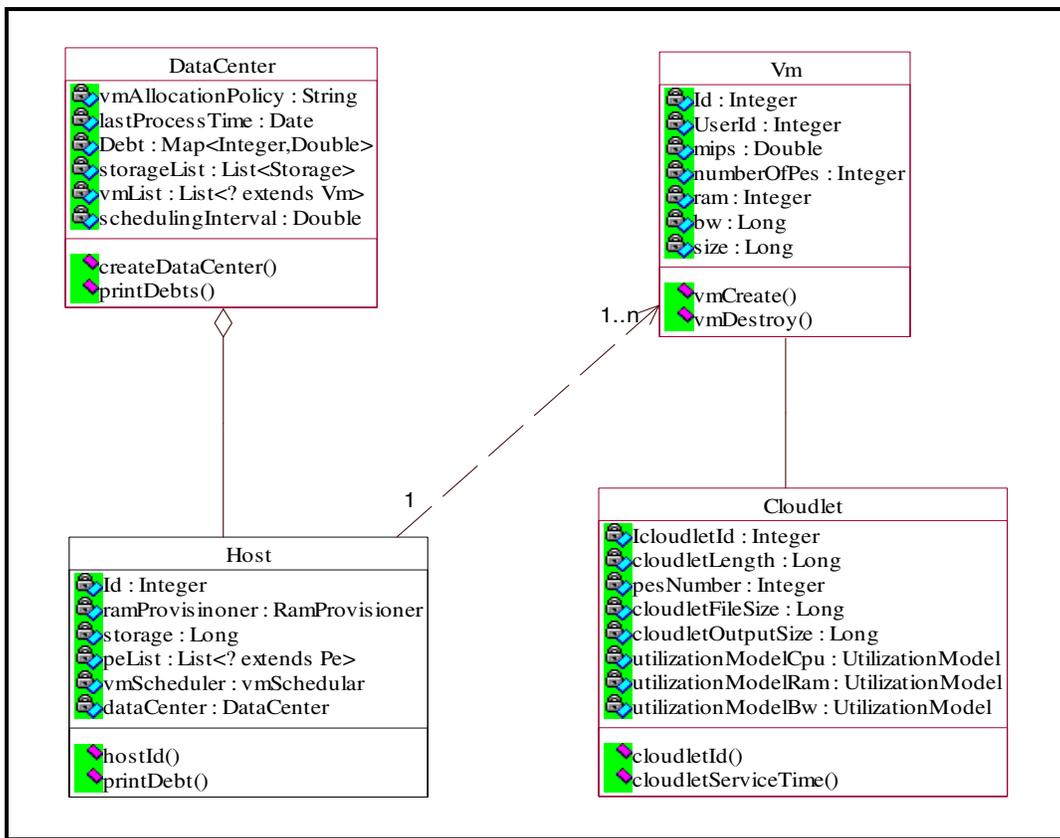


Figure 3: Class Diagram of CloudSim Components

8. RELATED WORK

Several algorithms have been developed and simulated in CloudSim environments for balancing and maintaining load in the cloud computing environment. Many of the existing researchers concentrate on virtual machine, hosts and response time specifically in order to balance load in cloud. These research works primarily focus on improving response time activity CloudSim environment mostly. In [21], Rodrigo presents an analysis for minimum amount of time and memory requirement to initialize an experiment while the hosts in the datacenter increases. The

result indicates that the time and memory requirement is linear. Rodrigo also verifies that the time increases exponentially when the number of hosts/VM is increased in the simulation environment. Warstein [19] emphasizes a model to find the suitable VM within very short period of time when any request arrives. He suggests that the least loaded VM would be selected to handle the request and the id of that VM would be sent to the datacenter controller for further information processing requirement. In [9], Kaur has discussed an algorithm called *active vm load balancer algorithm* to find the suitable VM in a short time period. He has stressed to count the maximum length of VM for the allocation of new request. If the length of the vm is not sufficient then a new VM would be added. After that all the VM's load needs to be counted and least loaded VM would be selected to handle the new request. Zhong deals with the issues in a simulation environment efficiently and significantly outperform the existing approaches through experimental results. Their results demonstrate that load balancing algorithm proposed in [25] can possibly improve the response time in order of magnitude with respect to number of VMs in Datacenter.

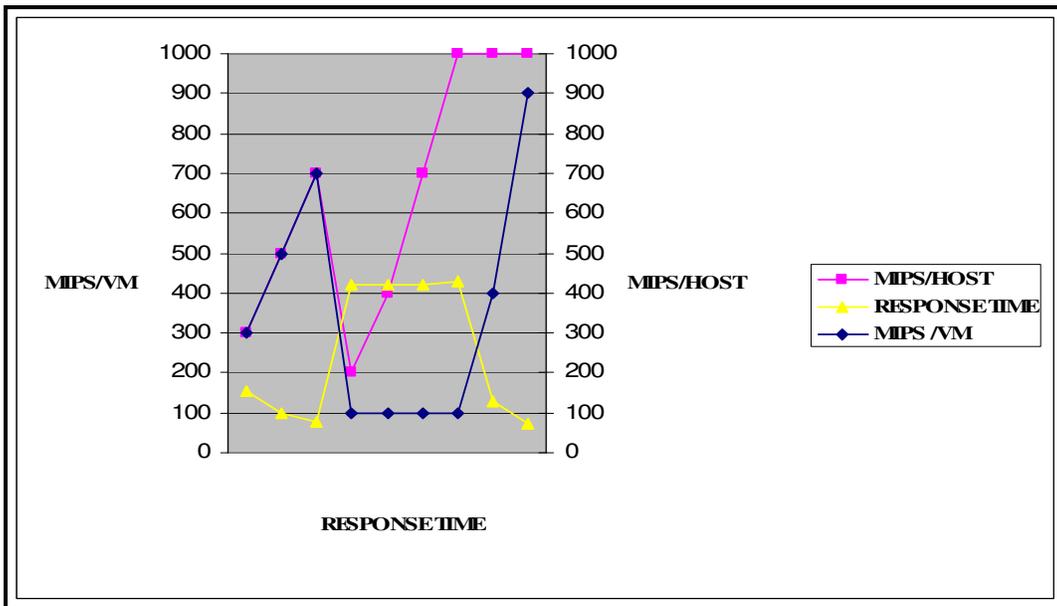


Figure 4: Variation of response time in round robin algorithm

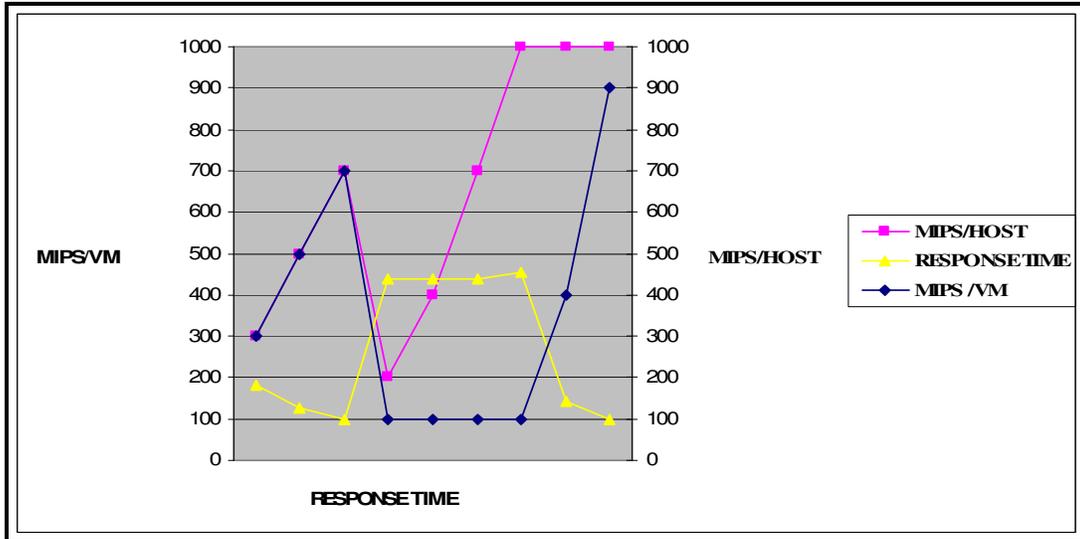


Figure 5: Variation of response time in central queuing algorithm

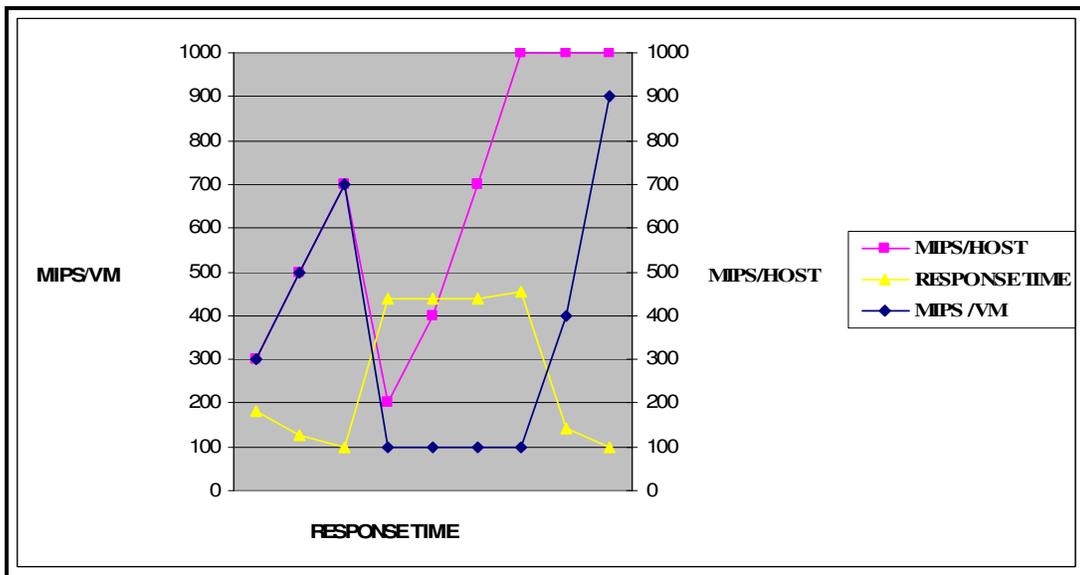


Figure 6: Variation of response time in randomized algorithm

9. CONCLUSION

This paper presents a concept of Cloud Computing along with research challenges in load balancing. It also focus on merits and demerits of the cloud computing. Major thrust is given on the study of load balancing algorithm, followed by a comparative survey of these above-mentioned algorithms in cloud computing with respect to stability, resource utilization, static or dynamicity, cooperative or non-cooperativeness and process migration. This paper aims towards the establishment of performance qualitative analysis on existing VM load balancing algorithm and then implemented in CloudSim and java language. Execution analysis of the simulation shows that change of MIPS will effect the response time. Increase in MIPS vs. VM decreases the response time.

It is observed with thorough study that, load balancing algorithm works on the principle on which situation workload is assigned, during compile time or run time. Depending on the compile time or run time it may be static or dynamic. Static algorithms are more stable than dynamic algorithm and it is easy to predict the behaviour of static algorithm also. Dynamic algorithms are really works better in case of distributed environments. Novel dynamic load balancing algorithm will be proposed and implemented as a future course of work. It is also necessary to propose a simulation model to evaluate the parameters or components in order to handle the random selection based load distributed problem.

REFERENCES:

- [1] Bhasker Prasad Rimal, Eummi Choi, Lan Lump (2009) "A Taxonomy and Survey of Cloud Computing System", 5th International Joint Conference on INC, IMS and IDC, IEEE Explore 25-27 Aug 2009, pp. 44-51
- [2] Bhatiya, Wickremasinghe.(2010)"Cloud Analyst: A Cloud Sim-based Visual Modeller for Analysing Cloud Computing Environments and Applications"
- [3] C.H.Hsu and J.W.Liu(2010) "Dynamic Load Balancing Algorithms in Homogeneous Distributed System," Proceedings of The 6th International Conference on Distributed Computing Systems, , pp. 216-223.
- [4] Calheiros Rodrigo N., Rajiv Ranjan, César A. F. De Rose, Rajkumar Buyya (2009): CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services CoRR abs/0903.2525: (2009)
- [5] Carnegie Mellon, Grace Lewis(2010) "Basics About Cloud Computing" Software Engineering Institute September
- [6] Cary Landis, Dan Blacharski, "Cloud Computing Made Easy", Version 0.3
- [7] CloudSim: A Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services, The Cloud Computing and Distributed Systems (CLOUDS) Laboratory, University of Melbourne, (2011) available from: <http://www.cloudbus.org/cloudsim/>
- [8] G. Khanna, K. Beaty, G. Kar, and A. Kochut,(2006) "Application Performance Management in Virtualized Server Environm.," in Network Operations and Management Symposium, (2006). NOMS (2006). 10th IEEE/IFIP, pp 373-381.
- [9] Jaspreet kaur (2012),"Comparison of load balancing algorithms in a Cloud" International Journal of Engineering Research and Applications(IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 3, pp.1169-1173
- [10] Java software version 7 downloaded from: <http://java.com/en/download/index.jsp>, August 2012.
- [11] Livny, M.; Melman, M. (2011): Load Balancing in Homogeneous Broadcast Distributed Systems. Proceedings of the ACM Computer Network: Performance Symposium, pp. 47-55.
- [12] M. Armbrust, A. Fox, R. Griffith, A. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, M. Zaharia (2009). Above the Clouds: A Berkeley View of Cloud Computing. Technical Report No. UCB/EECS-2009-28, University of California at Berkeley, USA, Feb. 10, 2009.
- [13] M. Beltran, A. Guzman and J.L. Bosque(2011), "Dealing with heterogeneity in clusters" in proceeding of the Fifth International Symposium on Parallel and Distributed Computing, ISPD.
- [14] Martin Randles, David Lamb, A. Taleb-Bendiab(2010), A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing IEE 24th International Conference on Advanced Information Networking and Applications Workshops.
- [15] Martin Randles, David Lamb, A. Taleb-Bendiab(2010),"A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing", IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, 20-23, pp. 551-556.
- [16] McEvoy, G. & Schulze, B(2008). "Using Clouds to address Grid Limitations," 1-6. Proceedings of the 6th International Workshop on Middleware for Grid Computing (MGC). Leuven, Belgium, December. ACM, 2008. ISBN: 978-1-60558-365-5
- [17] Michael Armbrust, Armando Fox, Gunho Lee, Ion Stoica(2009) "Above the Clouds :A Berkeley View of Cloud Computing" University of California at Berkeley Technical Report No. UCB/EECS-2009-28

- [18] Murata, Y. et al. (2010): A Distributed and Cooperative Load Balancing Mechanism for Large-Scale P2P Systems. Proceedings of International Symposium on Applications and Internet (SAINT '06) Workshops, pp. 126-129.
- [19] P. Warstein, H. Situ and Z. Huang (2010), "Load balancing in a cluster computer" In proceeding of the seventh International Conference on Parallel and Distributed Computing, Applications and Technologies, IEEE
- [20] Qingyang Meng et al. (2008): A Dynamic Load Balancing Method Based on Stability Analysis. Proceedings of International Symposium on Computer Science and Computational Technology (ISCST'08), pp. 404-408
- [21] Rodrigo N. Calheiros Rajiv Ranjan, César A. F. De Rose and Rajkumar Buyya (2011) : CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services Grid Computing and Distributed Systems (GRIDS) Laboratory Department of Computer Science and Software Engineering The University of Melbourne, Australia
- [22] Sandip Kumar Goyal, R.B. Patel, Manpreet Singh (2009) "Adaptive and Dynamic Load Balancing Methodologies For Distributed Environment" International Journal of Engineering Science and Technology (IJEST).
- [23] Wang, L., Tao, J., Kunze, M., Castellanos, A., Kramer, D., & Karl, W. (2008). "Scientific Cloud Computing: Early Definition and Experience," 825-830. Proceedings of the 10th IEEE International Conference on High Performance Computing and Communications (HPCC '08). Dalian, China, September 2008. IEEE
- [24] Zenon Chaczko, Venkatesh Mahadevan, Shahrzad Aslanzadeh, Christopher Mcdermid (2011) "Availability and Load Balancing in Cloud Computing" International Conference on Computer and Software Modeling IPCSIT vol.14 IACSIT Press, Singapore 2011
- [25] Zhong Xu, Rong Huang, (2009) "Performance Study of Load Balancing Algorithms in Distributed Web Server Systems", CS213 Parallel and Distributed Processing Project Report.

AUTHORS

Soumya Ray is presently working as a lecturer in Birla Institute of Technology, Mesra, Kolkata campus. He has completed his B.Tech from Kalyani Govt. Engineering College in 2006 and now pursuing M.Tech from Birla Institute of Technology, Mesra. His area of interest is distributed computing, specifically in cloud computing.



Ajanta De Sarkar is working as Associate Professor in the Department of CSE in Birla Institute of Technology, Mesra. She is having altogether 16 years of experience including 6 years of Industry experience. Having graduated from Bethune College, University of Calcutta in B.Sc. (Mathematics) in 1993, and completed MCA from Jadavpur University in 1996. She has been awarded PhD (Engg.) from Jadavpur University in 2009. Her field of Specialization is Distributed Computing, specifically Grid Computing. Her focused research area includes Grid Computing, Cloud Computing and Wireless Sensor Network.

