

Load Balancing Techniques in Cloud Computing

Swati Agrahari¹, Sapna Choudhary²

¹M.Tech Scholar, ²Prof. Dept. of CSE, SRGI, Jabalpur, India

swatiagrahari25@gmail.com, choudharysapnajain@gmail.com

Abstract-- Cloud computing is a rapidly expanding field of computing science and industry today, with numerous advantages including reduced time, limitless computing capacity, and versatile computing capabilities. It's a model that allows users to connect to a common pool of computing resources on demand. It encompasses a wide range of principles, including load balancing, scheduling, and so on. From the convergence and evolution of industry, technology, and methodology perspectives, cloud computing as computing resources developed as a new kind of service in the field of communication and information technology that enables and furnishes its users to access IT resources anywhere and anytime at low cost on a pay-per-use basis. The resources overcloud must be handled and controlled by an appropriate mechanism by setting the priorities on the available cloud resources which are shared among its users. To overcome these challenges, this paper conducts a systematic literature review on load balancing algorithms and challenges in cloud computing for one of the most important cloud resources, the Virtual Machine over Cloud environment.

Keywords: Cloud Computing, Load Balancing, Existing load balancing algorithms.

I. INTRODUCTION

Cloud computing is a new technology .It providing online resources and online storage to the user's .It provide all the data at a lower cost. In cloud computing users can access resources all the time through internet. They need to pay only for those resources as much they use .In Cloud computing cloud provider outsourced all the resources to their client. There are many existing issues in cloud computing. The main problem is load balancing in cloud computing. Load balancing helps to distribute all loads between all the nodes. It also ensures that every computing resource is distributed efficiently and fairly. It helps in preventing bottlenecks of the system which may occur due to load imbalance. It provides high satisfaction to the users. Load balancing is a relatively new technique that provides high resource utilization and better response time. [1] [2] [3] [4] Cloud computing provide many advantages to the users.

Cloud computing consist of several characteristestics: [5] [6].

- *On demand service*- Cloud computing provide services to users on their demand .Users can access the services as they want.
- *Broad Network Access*- In cloud computing capabilities are available over the network .All the capabilities are accessed through different mechanisms.

- *Resource Pooling*- Different models are used to pooled the resources which provide by the providers to their consumers. All the resources dynamically assigned and reassigned according to consumer demand.
- *Rapid Elasticity*- Quantity of resources is increase at any time according to the customer's requirements.
- *Measured Service*- In cloud computing resource usage can be monitored, controlled for both the provider and consumer of the all service.

Challenges in Cloud Computing

There are many challenges in cloud computing:-

1. Security
2. Efficient load balancing
3. Performance Monitoring
4. Consistent and Robust Service abstractions
5. Resource Scheduling
6. Scale and QoS management
7. Requires a fast speed Internet connection.

II. CLOUD COMPUTING MODEL

Hive is data warehouse software which is used for facilitates querying and managing large data sets residing in distributed storage. Hive language almost look like SQL language called HiveQL [4]. Hive is designed to enable easy data summarization. Hive also allows traditional map reduce programs to customize mappers and reducers when it is inconvenient or inefficient to execute the logic in HiveQL.

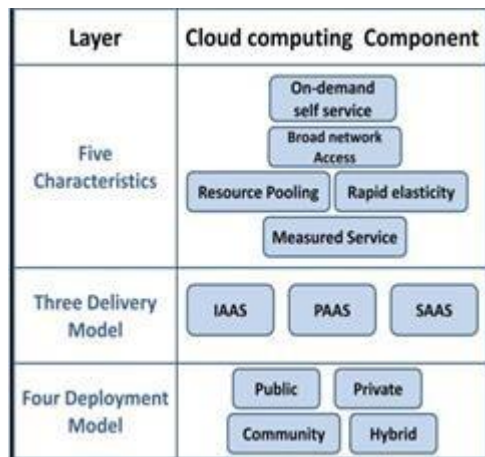
Fig: 1 shows Cloud computing model which consist services of cloud and different deployment models as:

Services of Cloud Computing:

Service means different types of applications provided by different servers across the cloud. There are many services are provide to the users over cloud. [7]

Software as a Service (SaaS): SaaS provided all the application to the consumer which are Provided by the providers. Applications are running on a cloud infrastructure. Interfaces (web browser) are used access the applications. The consumer does not control the internal function. [8] [9]

Platform as a Service (PaaS): PaaS provides all the resources to the customers that are required for building applications. It provides all the services on the internet. User not need to download and install the software.



Consumers deploy all the application onto the cloud infrastructure. There is different tools and programming languages are provided to the uses to develop the applications. The consumer does not control network, servers, operating systems, or storage. Consumer controls all applications which they deploy. Disadvantages There is very less portability among different providers.

Infrastructure as a Service (IaaS): In this service consumer does not manage or control the underlying cloud infrastructure. In infrastructure as a service consumer able to control operating systems, storage, and all applications which they deployed. There is a limited control of customer on the networking components. Infrastructure Providers control storing and processing capacity.

Virtualization is used assign and dynamically resizes these resources to build systems as demanded by customers. Consumers deploy the software stacks that run their services. Provider provide network, services as on demand services. User use these services directly .It can be used to avoid buying, housing , and managing the basic hardware and software infrastructure components, scales up and down quickly to meet demand.

III. VIRTUALIZATION

Virtualization means which are not exist in real, but it provides everything like real. Virtualization is the software implementation of a machine which will execute different programs like a real machine. Through the virtualization user can use the different applications or services of the cloud, so this is the main part of the cloud environment. There are different types of virtualization is used in cloud environment.

Two types of virtualization are:

1. Para virtualization

Full Virtualization: Full virtualization means a complete machine is installed on another machine. That virtual machine provides all the function which

exists on the original machine. It facilities when actual machine not free then user use the virtual machine.

Para virtualization: Para virtualization means the hardware allows multiple operating systems to run on single machine .It also allow efficient use of system resources such as memory and processor.

IV. LOAD BALANCING

Load balancing is used to distributing a larger processing load to smaller processing nodes for enhancing the overall performance of system. In cloud computing environment load balancing is required distribute the dynamic local workload evenly between all the nodes. . [10][11][12][13]

- Load balancing helps in fair allocation of computing resource to achieve a high
- User satisfaction and proper Resource utilization .High resource utilization and Proper load balancing helps in minimizing resource consumption. It helps in implementing fail over, scalability, and avoiding bottlenecks.
- Load balancing is a techniques that helped networks and resources by providing a Maximum through put with minimum response time. Load balancing is dividing the traffic between all servers, so data can be sent and received without any delay with load balancing.
- In cloud environment many algorithms are available that helps in proper traffic
- Loaded between all available servers. Most of them can be applied in the cloud environment with suitable verifications. In cloud computing environment load balancing algorithms can be divided into two main groups: first algorithm type is Batch mode heuristic scheduling algorithms (BMHA) and second is online mode heuristic algorithms. In BMHA Jobs are combined together when they are arriving in the system. The BMHA scheduling algorithm will start after a fixed time period.
- The examples of BMHA based algorithms are: First Come First Served Scheduling algorithm (FCFS), Round Robin scheduling algorithm (RR), Min Min algorithm and Max Min algorithm. In On-line mode heuristic scheduling algorithm, all Jobs are scheduled when they are arriving in the system. The cloud environment is a heterogeneous system and in this speed of each processor varies quickly and easily. The online mode heuristic scheduling algorithms are more appropriate and better for a cloud environment.

- It is very important to estimate proper load, need to do comparison of all load, stability of all different systems, performance of purposed system, interaction between all the nodes and nature of work to be transferred while developing a load balancing algorithm. The most important thing is selecting the nodes and its also include many other ones. CPU load, amount of memory required combine together to calculate the load of machine.
- In our daily life example of load balancing is websites. Users could experience many Problems without Load balancing like delays, timeouts and long system responses.

A. Load balancing classification:

Fig.3 represents different load balancing algorithms. This is mainly divided into two categories: static load balancing algorithm and dynamic load balancing algorithm:

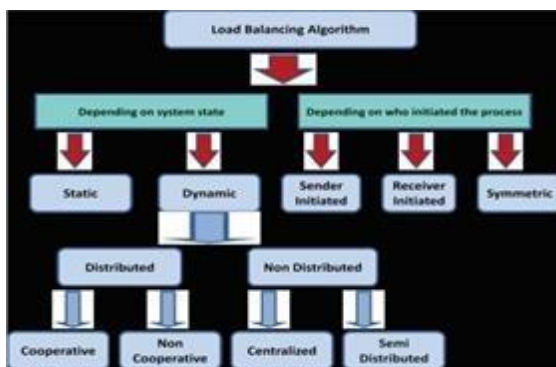


Fig.3 (Types of load balancing algorithm)

- 1) *Static approach*: - This approach is mainly defined in the design or implementation of the system. Static load balancing algorithms divide the traffic equivalently between all servers.
- 2) *Dynamic approach*: - This approach considered only the current state of the system during load balancing decisions. Dynamic approach is more suitable for widely distributed systems such as cloud computing.

Dynamic load balancing approaches have two types. They are distributed approach and non-distributed (centralized) approach. It is defined as following:

- a) *Centralized approach*: - In centralized approach, only a single node is responsible for managing and distribution within the whole system. Other all nodes are not responsible for this.
- b) *Distributed approach*: - In distributed approach, each node independently builds its own load vector. Vector collecting the load information of other nodes. All decisions are made locally using local load vectors. Distributed approach is more suitable for widely distributed systems such as cloud computing.

B. Metrics for Load Balancing:

1. *Throughput*: - It is used to calculate the all tasks whose execution has been completed. The performance of any system is improved if throughput is high.
2. *Fault Tolerance*: - It means recovery from failure. The load balancing should be a good fault-tolerant technique.
3. *Migration time*: - It is the time to migrate the jobs or resources from one node to other nodes. It should be minimized in order to enhance the performance of the system.
4. *Response Time*: - It is the amount of time that is taken by a particular load balancing algorithm to response a task in a system. This parameter should be minimized for better performance of a system.
5. *Scalability*: - It is the ability of an algorithm to perform Load balancing for any finite number of nodes of a system. This metric should be improved for a good system.

C. Policies of load balancing algorithm

There are many policies are used in load balancing algorithms: [14] [15]

- *Information policy*: It defined that what information is required and how this information is collected. This is also defined that when this information is collected
- *Triggering policy*: This policy defined that time period when the load balancing operation is starting to manage the load.
- *Resource type policy*: This policy defined the all types of resources which are available during the load balancing.
- *Location policy*: This uses all the results of the resource type policy. It is used to find a partner for a server or receiver.
- *Selection policy*: This policy is used to find out the task which transfers from overloaded node to free node.

D. Major goals of load balancing algorithms

1. *Cost effectiveness*: Load balancing help in provide better system performance at lower cost.
2. *Scalability and flexibility*: The system for which load balancing algorithms are implemented may be change in size after some time. So the algorithm must handle these types' situations. So algorithm must be flexible and scalable.
3. *Priority*: Prioritization of the resources or jobs needs to be done. So higher priority jobs get better chance to execute.

V. EXISTING LOAD BALANCEING ALGORITHMS

There are many load balancing algorithms which help to achieve better throughput and improve the response time in cloud environment. All the algorithms have their own benefits. [16] [17] [18]

1. *Task Scheduling based on LB*: This algorithm mainly consist two level task scheduling mechanism which are based on load balancing to meet dynamic requirements of users. It obtains high resource utilization. This algorithm achieves load balancing by first mapping tasks to virtual machines and then all virtual machines to host resources. It is improving the task response time. It also provide better resource utilization.
2. *Opportunistic Load Balancing*: OLB is to attempt each node keep busy, therefore does not consider the present workload of each computer. OLB assigns each task in free order to present node of useful. The advantage is quite simple and reach load balance but its shortcoming is not consider each expectation execution time of task, therefore the whole completion time (Make span) is very poor.
3. *Round Robin*: - In this algorithm all the processes are divided between all processors. In this each process is assigned to the processor in a round robin order. The work load distributions between processors are equal. Different processes have not same job processing time. At many point of time some nodes may be heavily loaded and others remain idle. In web servers where http requests are of similar nature and distributed equally then RR algorithm is used. In Round Robin Scheduling the time quantum play a important role. When time quantum is very large then RR Scheduling Algorithm is same as the FCFS Scheduling, and when time quantum is too small then Round Robin Scheduling is known as Processor Sharing Algorithm.
4. *Randomized*: This algorithm is static in nature. In this algorithm a process can be handled by a particular node n with a probability p . When all the processes are of equal loaded then this algorithm work well. Problem arises when loads are of different computational complexities. This algorithm is not maintaining deterministic approach.
5. *Min-Min Algorithm*: It starts with a set of all unassigned tasks. In this minimum completion time for all tasks is found. Then after that among these minimum times the minimum value is selected. Then task with minimum time schedule on machine. After that the execution time for all other tasks is updated on that machine then again the same procedure is followed until all the tasks are assigned on the resources. The main problem of this algorithm is has a starvation.
6. *Max-Min Algorithm*: Max-Min algorithm is almost same as the min-min algorithm. The main difference is following: In this algorithm first finding out minimum execution times, then the maximum value is selected which is the maximum time among all the tasks on any resources. After that maximum time finding, the task is assigned on the particular selected machine. [19] Then the execution time for all tasks is updated on that machine, this is done by adding the execution time of the assigned task to the execution times of other tasks on that machine. Then all assigned task is removed from the list that executed by the system.
7. *Honeybee Foraging Behavior*: It is a nature inspired Algorithm for self-organization. Honeybee achieves global load balancing through local server actions. The performance of the system is enhanced with increased system diversity. The main problem is that throughput is not increased with an increase in system size. When the diverse population of service types is required then this algorithm is best suited.
8. *Active Clustering*: - In this algorithm same type nodes of the system are grouped together and they work together in groups. It works like as self- aggregation load balancing technique where a network is rewired to balance the load of the system. Systems optimize using similar job assignments by connecting similar services. System Performance improved with improved resources. The throughput is improved by using all these resources effectively.
9. *Compare and Balance*: - This algorithm is uses to reach an equilibrium condition and manage unbalanced systems load. In this algorithm on the basis of probability (no. of virtual machine running on the current host and whole cloud system), current host randomly select a host and compare their load. If load of current host is more than the selected host, it transfers extra load to that particular node. Then each host of the system performs the same procedure. This load balancing algorithm is also designed and implemented to reduce virtual machines migration time. Shared storage memory is used to reduce virtual machines migration time.
10. *Lock-free multiprocessing solution for LB*: It proposed a lock-free multiprocessing load balancing solution that avoids the use of shared memory in contrast to other multiprocessing load balancing solutions which use shared memory and lock to maintain a user session. It is achieved by modifying kernel. This solution helps in improving the overall performance of load balancer in a multicore environment by running multiple load-balancing processes in one load balancer.

11. *Ant Colony Optimization*:- Ant algorithms is a multiagent approach to difficult combinatorial optimization problems. Example of this approach is travelling salesman problem (TSP) and the quadratic assignment problem (QAP). These algorithms were inspired by the observation of real ant colonies. Ant's behaviour is directed more to the survival of the colonies. They not think for individual.

VI. CONCLUSION

Cloud computing mainly deals with software, data access and storage services that may not require end-user knowledge of the physical location and configuration of the system that is delivering the services. In the cloud storage, load balancing is a key issue. It helps in proper utilization of resources and hence in enhancing the performance of the system. A few existing algorithms can maintain load balancing and provide better strategies through efficient scheduling and resource allocation techniques as well. This paper presents a concept of Cloud Computing along with load balancing. Main thing is considered in this is load balancing algorithm. There are many above mentioned algorithms in cloud computing which consist many factors like scalability, better resource utilization, high performance, better response time.

REFERENCES

- [1] Mell, Peter and Grance, Tim, "The NIST definition of cloud computing", National Institute of Standards and Technology, 2009, vol 53, pages 50, Mell 2009
- [2] <http://www.ancoris.com/solutions/cloudcomputing.html>, "Cloud Computing | Google Cloud - Ancoris."
- [3] <http://www.personal.kent.edu/~rmuhamma/OpSystems/Myos/prioritySchedule.htm>, "Priority Scheduling - Operating Systems Notes."
- [4] S. S. MOHARANA, R. D. RAMESH, D. POWAR, "Analysis of load balancers in cloud computing" International Journal of Computer Science and Engineering, 2013, vol 2, pages 101-108
- [5] <http://blog.nexright.com/?cat=6> "Cloud Computing « Nexright Blog". A. Sidhu, S. Kinger, "Analysis of load balancing techniques in cloud computing", INTERNATIONAL JOURNAL OF COMPUTERS & TECHNOLOGY 4 (2) (2013) pages 737- 741.
- [6] <http://letslearncloud.wordpress.com/> "Cloud Computing | Learn Cloud and its tips."
- [7] Sahu, Yatendra and Pateriya, RK, "Cloud Computing Overview with Load Balancing Techniques", International Journal of Computer Applications, 2013, vol. 65, Sahu 2013
- [8] Chaudhari, Anand and Kapadia, Anushka, "Load Balancing Algorithm for Azure Virtualization with Specialized VM", 2013, algorithms, vol 1, pages 2, Chaudhari
- [9] Nayandeep Sran, Navdeep Kaur, "Comparative Analysis of Existing Load Balancing Techniques in Cloud Computing", vol 2, jan 2013
- [10] Bala, Anju and Chana, Inderveer, "A survey of various workflow scheduling algorithms in cloud environment", 2nd National Conference on Information and Communication Technology (NCICT), 2011
- [11] Chaczko, Zenon and Mahadevan, Venkatesh and Aslanzadeh, Shahrzad and Mcdermid, Christopher, "Availability and load balancing in cloud computing", International Conference on Computer and Software Modeling, Singapore, chaczko2011availability
- [12] <http://www.writeaprogram.info/c/os-programs/priority-scheduling/> "Priority Scheduling Algorithm Example - Write a Program."
- [13] R. Alonso-Calvo, J. Crespo, M. Garcia-Remesal, A. Anguita, V. Maojo, "On distributing load in cloud computing: A real application for very- large image datasets", Procedia Computer Science 1 (1) (2010) pages 2669-2677.
- [14] S.-C.Wang, K.-Q. Yan, S.-S.Wang, C.-W. Chen, "A three-phases scheduling in a hierarchical cloud computing network", in: Communications and Mobile Computing (CMC), 2011 Third International Conference on, IEEE, 2011, pp. 114-117.
- [15] O. Elzeki, M. Reshad, M. Elsouad, "Improved max- min algorithm in cloud computing, International Journal of Computer Applications" vol 50 (12) (2012) pages 22-27..
- [16] U. Bhoi, P. N. Ramanuj, "Enhanced max-min task scheduling algorithm in cloud computing", International Journal of Application or Innovation in Engineering and Management (IJAEM), ISSN, 2013, pages 2319-4847
- [17] H. Chen, F. Wang, N. Helian, G. Akanmu, "User-priority guided min- min scheduling algorithm for load balancing in cloud computing", Parallel Computing Technologies (PARCOMPTECH), 2013 National Conference on, 2013
- [18] Y. Hu, R. Blake, D. Emerson, "An optimal migration algorithm for dynamic load balancing", Concurrency: Practice and Experience 10 (6) (1998) pages 467-483.