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

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 5, Issue 5, May 2017

A Comparative Survey on Load Balancing Algorithm in Cloud

Surbhi Nema[1], Prof. Saurabh Kapoor[2]

[1]Research Scholar, Dept. of Computer Science, Gyan Ganga Institute of Technology and Science, Jabalpur, Madhya Pradesh, India

[2]Assistant Professor, Dept. of Computer Science, Gyan Ganga Institute of Technology and Science, Jabalpur, Madhya Pradesh, India

ABSTRACT: Cloud computing is the delivery of computing services—servers, storage, databases, networking, software, analytics and more—over the Internet ("the cloud"). Companies offering these computing services are called cloud providers and typically charge for cloud computing services based on usage, similar to how you are billed for water or electricity at home.. The main challenge of cloud computing is load balancing to maximize throughput. This paper has survey of different load balancing algorithms and their comparison with each other.

KEYWORDS*:* Cloud computing, Load balancing, Job Scheduling.

I. INTRODUCTION

Cloud computing is a fast growing area in computing research and industry today. Three main services provided by the cloud are IaaS, SaaS, and PaaS. With the advancement of the Cloud, there are new possibilities opening up on how applications can be built and how different services can be offered to the end user through Virtualization, on the internet. There are the cloud services providers who provide large scaled computing infrastructures defined on usage, and provide the infrastructure services in a very flexible manner. The establishment of an effective load balancing algorithm and how to use.Generally it consists of a bunch of distributed servers known as masters, providing demanded services and resources to different clients known as clients in a network with scalability and reliability of datacenter. The distributed computers provide on-demand services and Cloud computing resources efficiently for effective and efficient cloud computing as is one of the Clouds computing service provider's ultimate goals.

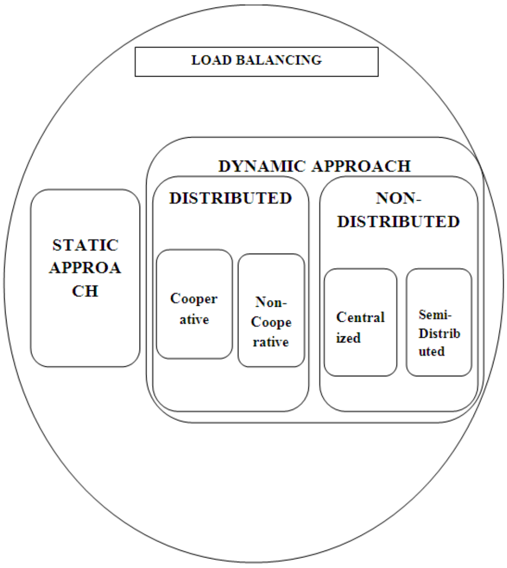
Load balancing distributes workloads across multiple computing resources, such as computers, a computer cluster, network links, central processingunits or disk drives. Load balancing is essential for efficient operations in distributed environments. It means distributing the amount of work to do between different servers in order to get more work done in the same amount of time and serve clients faster. Load balancing aims to optimize resource use, maximize throughput, minimize response time, and avoid overload of any single resource. Using multiple components with load balancing instead of a single component may increase reliability and availability through redundancy. Jobs cannot be assigned to appropriate servers and clients individually for efficient load balancing as cloud is a very complex structure and components are present throughout a wide spread area. Load balancing enhances the overall performance of the system. It provides high resource utilization and better response time [1]. In cloud computing load balancing is required to do the distribution of dynamic local workload evenly between all the Virtual Machines. [2]

DOI:10.15680/IJIRCCE.2017. 0505250

1.1 Load Balancing Classification

The below figure illustrates the types of load balancing algorithms. Mainly they are divided into two categories:

(1) Static Approach
(2) Dynamic Approach



Static approach: **-** Static load balancing algorithm equally divides the traffic between all the users. The only information that it uses is the average behavior of the system. This approach ignores the current state or the load of the node in the system.

Dynamic approach:**-**This approach takes into consideration only the current state of the system during load balancing decision. It is more suitable for widely distributed system such a cloud computing. Also this approach overcomes the difficulties of static approach.

Dynamic approach can be categorized into two parts

- Centralized Approach: Only a single node is responsible for managing and distribution within the whole system.

- Distributed Approach: Each node independently builds its own load vector. Vector collecting load information of other node. All decision is made locally using local load vector.

1.2 Load balancing Policies

Load balancing algorithms can be defined based on the implementation of the following policies: [3]

- Information Policy: This policy defines what is the amount of information is required and how this information can be collected by different means. It also defines when this information can be collected.
- Triggering Policy: This policy defines the time period when the load balancing operation is starting to manage the load as a precaution to eliminate a collapse.
- Resource type Policy: This policy defines all the types of different resources which are available

during the load balancing.

- Location Policy: This policy uses all the results of the resource type policy. It is used to find a partner for a server or a receiver across the domain.
- Selection Policy: This policy is used to find out the task which is transfered from overloaded node to free node.

1.1 Metrics for load balancing

1. Throughput:It is used to the all task 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 tolerance 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 the 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.

II. RELATED WORKS

In this section we are going to analyze different algorithms for load balancing.

A.) Round Robin Algorithm: Round Robin Algorithm [4] proposed by Antony Thomas is a default load balancing method which passes each new connection request to the next server in line, and thereby eventually distributing connections, evenly across the array of machines being considered for load balancing. Round Robin mode works well in configurations where the equipment that you are load balancing is roughly equal in processing speed and memory

B.) Throttled Load Balancing Algorithm:- Nitika [5] proposed a throttled load balancing algorithm which reduces cost and response time across VM's in multi data center and optimizes response time. In this paper author described algorithm in which the client first requests the load balancer to find a suitable virtual machine to perform the required operation for the incoming process. In cloud computing, there may be multiple instances of virtual machine. These virtual machines can be grouped based on the type of request they handled. So as per the incoming requests it works accordingly. Whenever a client sends a request, the load balancer will first look for the group and if it is ready to accept and handle the request it is going to assign request to it.

C.) Min Min Load Balancing Algorithm:- Kokilavani proposed a Min-Min algorithm[6] which selects the task which has maximum completion time and assigns it to appropriate resource to get a better makespan and utilize the resource. When the number of the small tasks is more than the number of the large tasks in a meta-task, the Min-Min algorithm cannot schedule tasks, appropriately, and the makespan of the system gets relatively large. Furthermore it does not provide a load balanced schedule. This algorithm gives higher priority to small tasks and it increases the response time for large tasks. Thus the drawback of this algorithm is that some jobs may experience starvation.

D.) Stochastic Hill Climbing Algorithm:-Brototi Mondal and Kausik Dasgupta proposed a stochastic hill climbing algorithm [7] which is used for allocating incoming jobs to servers or virtual machines. Performance of the algorithm is analyzed qualitatively and quantitatively using Cloud Analyst. Cloud Analyst is a cloudsim based

Visual Modeller for analyzing cloud computing environments and applications. In this paper the author describes this approach has been used for load distribution in Cloud computing environment. The disadvantage of this approach is it doesn't obtain a fair optimized makespan in many cases.

E.) Fuzzy Active Monitoring Load Balancing Algorithm:- Srinivas Sethi[8] proposed an efficient load balancing in cloud computing using fuzzy logic. This algorithm uses two parameters namely processor speed and load on virtual machine. The authors have introduced a new fuzzy logic based dynamic load balancing algorithm with additional parameters like memory usage, bandwidth usage, disk space usage, virtual machine status. The processor speed and assigned load of Virtual Machine that are used to balance the load in cloud computing.

F.) Genetic Algorithm: - Chandrashekaran K [9] proposed a genetic algorithm which aims to design and implement an algorithm for a scheduling strategy on Virtual Machine Resources in cloud computing environment using current system state such that it achieves load balancing and hence Virtual Machine migration problem optimization. It will compute the influence on system after deployment of Virtual Machine resources before actually deploying and then selects the best solution having the least load imbalance. But they have not considered about the power consumption and cost during the implementation

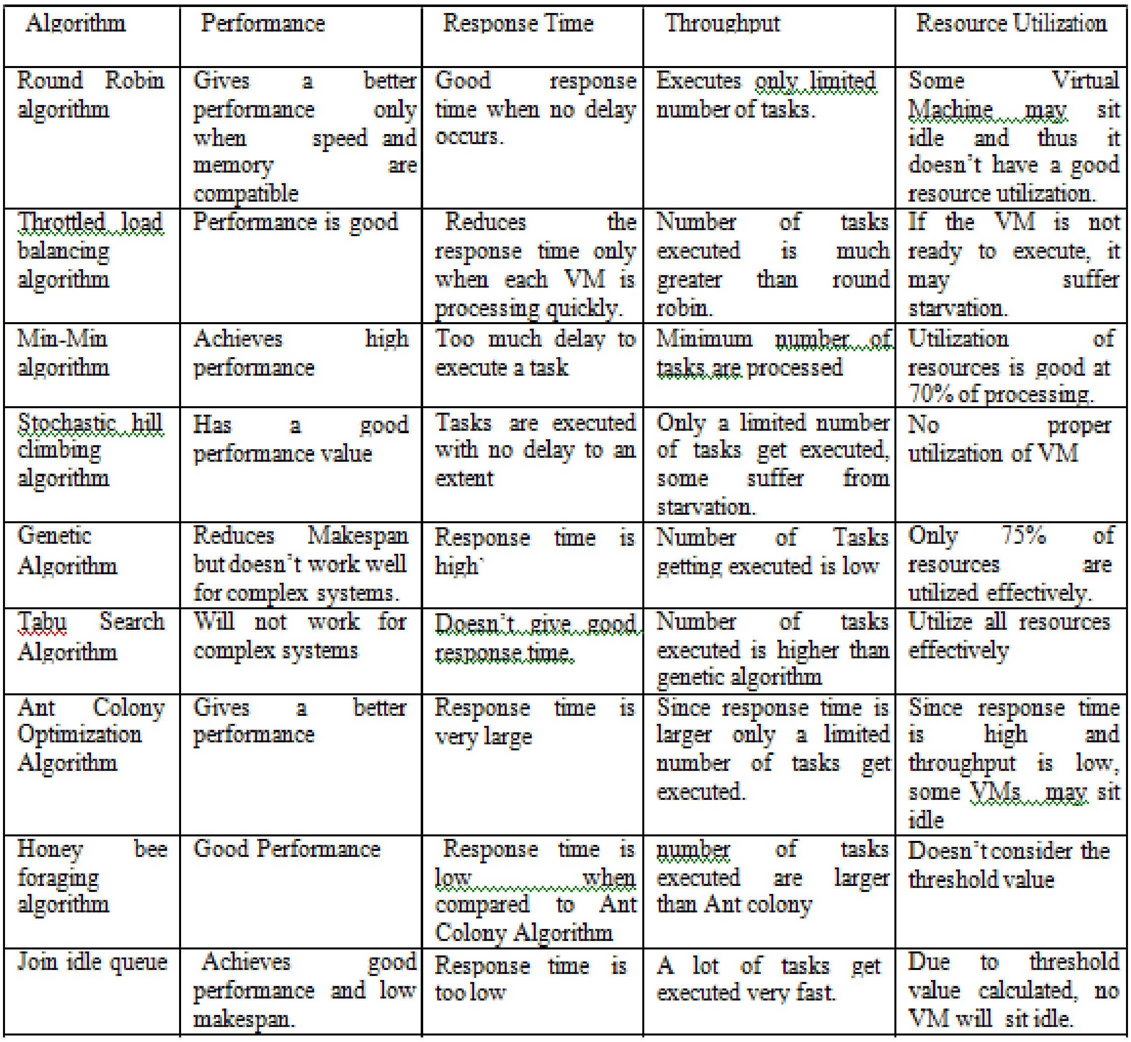
G.) Tabu Search Algorithm:-Kousik Dasgupta [10] proposed an algorithm named Tabu Search which assigns the workflow tasks to multiple resources to make the total cost of execution as minimal and the load to be shared among these computing resources. Tabu search algorithm is one of the most popular local search techniques based on the neighborhood search algorithm. This can be directly applied to virtually any kind of optimization problem. This algorithm is used to solve scientific workflow scheduling application in cloud computing. The workflow problem considered is similar in structure to a version of the Evolutionary Multi-Objective Optimization Problem (EMO). But this algorithm doesn't provide a better performance and can't be used in complex workflow scheduling applications.

H.) Ant Colony Optimization Algorithm: This algorithm proposed by Marco Dorigo [11] works best in scenario where the under loaded node is found at the beginning of the search. It gives optimal result. Ant Colony Optimization (ACO) is a population-based approach for solving combinatorial optimization problems that is inspired by the foraging behavior of ants and their inherent ability to find the shortest path from a food source to their nest. ACO Algorithm has been applied to a broad range of hard combinatorial problems. Among them, we have the classic Traveling Salesman Problem (TSP), where an individual must find the shortest route by which to visit a given number of destinations. ACO is inspired from the ant colonies that work together in foraging behavior. It works efficiently and achieves better utilization of resources, but gives a relatively more overhead during runtime.

I.) Honey Bee foraging algorithm :[12] This algorithm is derived from the behavior of honey bees for finding and reaping food. In order to check for fluctuation in demand of services, servers are grouped under virtual servers, having its own virtual service queues. Each server processing a request from its queue calculates a profit or reward on basis of CPU utilization, which it corresponds to the quality that the bees show in their waggle dance and advertise on the advert board. Each of the servers takes the role of either forager or scout behavior of bee. This algorithm is very much analogous to the behavior of how honey bees finds and reaps their food. There is a category of bees called forager bees. They search for food and after getting it they come back for announcement. They announce it by doing a dance called waggle dance. This dance is the description of available metadata food. After getting the information the scout bees follows the searcher bees towards the food location for storage purpose. Then returning to beehive they again do a waggle dance which gives the information of available food to be occupied and then more food can be consumed by the honey bee. In load balancing with the increasing and decreasing web server's demand, the services are also assigning dynamically to map the users changing demands. Within virtual servers the server are clustered, each virtual server having its own virtual service queue. Like the quality that bee shows by their waggle dance each server also calculates a profit or reward from the request queues based on the optimization criteria. This reward can be measured by the amount of time that the CPU spends on the processing of a request. In case of honey bees the dance floor is analogous to an advert board here. The disadvantage of this algorithm is that it completely ignores the idle time of Virtual Machine.

J.) Join Idle Queue Algorithm(JIQ):This algorithm proposed by Yi Lu[13]is a novel load balancing algorithm in large systems and it effectively utilizes the idle time of virtual machine. It has a good response time, high throughput and high resource utilization. This algorithm incurs no communication cost between dispatchers and processors at job arrivals.Also it doesn't make any Virtual Machine to sit idle. Thus Idle time of Virtual Machines are effectively utilized. Join Idle queue algorithm is tested at different levels of threshold value for an efficient load balancing.

III. COMPARATIVE ANALYSIS



Algorithm	Performance	Response Time	Throughput	Resource Utilization
Round Robin algorithm	Gives a better performance only when speed and memory are compatible	Good response time when no delay occurs.	Executes only limited number of tasks.	Some Virtual Machine may sit idle and thus it doesn't have a good resource utilization.
Throttled load balancing algorithm	Performance is good	Reduces the response time only when each VM is processing quickly.	Number of tasks executed is much greater than round robin.	If the VM is not ready to execute, it may suffer starvation.
Min-Min algorithm	Achieves high performance	Too much delay to execute a task	Minimum number of tasks are processed	Utilization of resources is good at 70% of processing.
Stochastic hill climbing algorithm	Has a good performance value	Tasks are executed with no delay to an extent	Only a limited number of tasks get executed, some suffer from starvation.	No proper utilization of VM
Genetic Algorithm	Reduces Makespan but doesn't work well for complex systems.	Response time is high`	Number of Tasks getting executed is low	Only 75% of resources are utilized effectively.
Tabu Search Algorithm	Will not work for complex systems	Doesn't give good response time.	Number of tasks executed is higher than genetic algorithm	Utilize all resources effectively
Ant Colony Optimization Algorithm	Gives a better performance	Response time is very large	Since response time is larger only a limited number of tasks get executed.	Since response time is high and throughput is low, some VMs may sit idle
Honey bee foraging algorithm	Good Performance	Response time is low when compared to Ant Colony Algorithm	number of tasks executed are larger than Ant colony	Doesn't consider the threshold value
Join idle queue	Achieves good performance and low makespan.	Response time is too low	A lot of tasks get executed very fast.	Due to threshold value calculated, no VM will sit idle.

IV. CONCLUSION

In this paper we have pondered in depth about load balancing algorithms for cloud. We studied and analyzed various algorithms on load balancing and compared them on the basis of performance issues, resource utilization, response time and throughput for different workflows. In this detailed study we found that the Join Idle queue algorithm shows a better efficient optimization in terms of performance, resource utilization, throughput among these algorithms. In the future we have planned to use Join Idle Queue algorithm with different threshold values and communication cost for better optimization results in cloud scheduling.

REFERENCES

[1] Rajkumar Buyya, Chee Shin Yeo, Srikumar Venugopal,James Broberg, and Ivona Brandic, "Cloud Computing and Emerging IT Platforms: Vision, Hype, and Reality for Delivering Computing as the 5th Utility, Future Generation Computer Systems", Volume 25, Number 6, pp: 599-616, ISSN:0167-739X, Elsevier Science, Amsterdam, The Netherlands, June 2009.
[2] Chaudhari, Anand and Kapadia, Anushka, "Load Balancing Algorithm for Azure Virtualization with Specialized VM", 2013, algorithms, Volume1, pp.201-203.
[3] 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), pp. 2669–2677.
[4] Antony Thomas, Krishnalal G,A" Novel Approach of Load Balancing Strategyin Cloud Computing" InternationalConference On Innovations & Advances In Science, Engineering And Technology [IC - IASET 2014] Volume 3, Special Issue 5, July 2014.
[5] Ms..Nitika, Ms.Shaveta, Mr. Gaurav Raj; "ComparativeAnalysis of Load Balancing Algorithms inCloud Computing", International Journal ofAdvanced Research in Computer Engineering &Technology Volume 1, Issue 3, May 2012.
[6] T. Kokilavani and Dr. D.I. George Amalarethinam,"Load Balanced Min-Min Algorithm forStatic Meta- Task Scheduling in Grid Computing"InternationalJournal of Computer Applications Volume 20– No.2,pp.0975-8887, April 2011.
[7] Brototi Mondal, Kousik Dasgupta, Paramartha Dutta,"Load Balancing in Cloud Computing using Stochastic Hill Climbing-A Soft Computing Approach", Procedia Technology 4 (2012), pp.783 –789.
[8] S. Sethi, A. Sahu, and S. K. Jena, "Efficient load balancing in cloud computing using fuzzy logic," IOSR Journal of Engineering, vol. 2, no. 7, pp.65–71, 2012.
[9] Chandrasekaran K., Usha Divakarla,"Load Balancing of Virtual Machine Resources in Cloud UsingGenetic Algorithm", Elsevier Publications 2013.