Comparative Analysis of various Load Balancing Algorithm

Prof. Nikhil S. Band
Information Technology, PRMIT&R, Badnera, India

Mr. Pravin R. Nerkar
Information Technology, PRMIT&R, Badnera, India

Mr. Rahul R. Papalkar
Information Technology, PRMIT&R, Badnera, India

Abstract—Cloud computing (CC) almost all IT resources can be delivered as a cloud service: applications, compute power, storage capacity, networking, programming tools, even communications services and collaboration tools. In other word CC defines the data is mostly resident on servers ‘somewhere on the Internet’ and the application runs on both the ‘cloud servers’ and the user’s browser. Load balancing is a technique to distribute workload across multiple computers, or other resources over the network links to achieve optimal resource utilization, minimum data processing time, minimum average response time, and avoid overload. Load balancing ensures that all the processors in the system as well as in the network does approximately the equal amount of work at any instant of time. Load Balancing is important for essential operations in cloud computing environments.

As cloud computing has been growing rapidly and many clients all over the world are demanding more services and better results, so load balancing is an essential and important research area. Thus Paper describe The Study and Analysis of various Load Balancing Algorithm use in the Cloud computing Environment along with their corresponding advantages, disadvantages and performance metrics are discussed in detail.

Keywords:- Cloud computing (CC), Load balancing.

I. INTRODUCTION

In many ways, cloud computing is simply a metaphor for the Internet, the increasing movement of compute and data resources onto the Web. But there’s a difference: cloud computing represents a new tipping point for the value of network computing. It delivers higher efficiency, massive scalability, and faster, easier software development. It’s about new programming models, new IT infrastructure, and the enabling of new business models.

At a basic level, cloud computing is simply a means of delivering IT resources as services. Almost all IT resources can be delivered as a cloud service: applications, compute power storage capacity, networking, programming tools, even communications services and collaboration tools. Cloud computing began as large-scale Internet service providers such as Google, Amazon etc. An architecture emerged: massively scaled, horizontally distributed system resources, abstracted as virtual IT services and managed as continuously configured, pooled resources. This architecturally similar to grid computing, but where grids are used for loosely coupled, technical computing applications, this new cloud model was being applied to Internet services. In this architecture, the data is mostly resident on servers ‘somewhere on the Internet’ and the application runs on both the ‘cloud servers’ and the user’s browser. In another word Cloud computing is an on demand service in which shared resources, information, software and other devices are provided according to the client’s requirement at specific time[1].

Architecture of cloud computing [14]

There are four types of cloud deployment model in,

- Private Cloud(used by single organization)
- Public Cloud(anyone can access)
- Community Cloud(shared by several organizations)
- Hybrid Cloud(combination of two or more clouds)

According to the National Institute of Standards and Technology (NIST), basic services provided by the Cloud Environment are as below [2]:-

- Software as a Service (SaaS): Customers hire software hosted by vendor.
- Platform as a Service (PaaS): Customers hire infrastructure and programming tools hosted by vendor to create applications.
- Infrastructure as a Service (IaaS): Customers hire processing, networking, storage and other fundamental computing resources.

National Institute of Standards and Technology (NIST) defines characteristics of Cloud Computing are as below [2]:-
II. VIRTUALIZATION

Virtualization is a cornerstone design technique for all cloud architectures. In cloud computing it refers primarily to platform virtualization or the abstraction of physical IT resources from the people and applications using them. Virtualization allows servers, storage devices, and other hardware to be treated as a pool of resources rather than discrete systems, so that these resources can be allocated on demand. In cloud computing, we’re interested in techniques such as par virtualization, which allows a single server to be treated as multiple virtual servers, and clustering, which allows multiple servers to be treated as a single server. There are three types of virtualization in cloud computing.

Operating System Virtualization: The use of OS-level virtualization or partitioning in cloud architectures can help solve some of the core security, privacy, and regulatory issues that could otherwise hinder the adoption of cloud computing.

Platform virtualization: Platform virtualization allows arbitrary operating systems and resulting application environments to run on a given system. There are two basic models for this system virtualization: full virtualization, or a complete simulation of underlying hardware, and paravirtualization, which offers a “mostly similar” model of the underlying hardware these are implemented as Type 1 hypervisors, which run directly on hardware, and Type 2 hypervisors, which run on top of a traditional operating system.

Application Virtualization: There is also a software angle to “containers” within the cloud. The Web container technology implemented in the cloud greatly impacts developer productivity and flexibility. The Web container is the part of the application server that manages servlets, JavaServer Page (JSP) files, and other Web-tier components.

III. LOAD BALANCING

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 resource 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.

The aim of load balancing is as follows:

- To increase the availability of services.
- To increase the user satisfaction.
- To maximize resource utilization.
- To reduce the execution time and waiting time of task coming from different location.
- To improve the performance.

- Maintain system stability.
- Build fault tolerance system.
- Accommodate future modification.

IV. CHALLENGES OF LOAD BALANCING

Overhead Associated: determines the amount of overhead involved while implementing a load-balancing system. It is composed of overhead due to movement of tasks, inter-process communication. Overhead should be reduced so that a load balancing algorithm performs well.

Throughput: It is the number of task executed in the fixed interval of time. To improve the performance of the system, throughput should be high.

Performance: It can be defined as the efficiency of the system. It must be improved Resource Utilization is used to test the utilization of resources. It should be maximum for an efficient load balancing system.

Scalability: The quality of service should be same if the number of users increases. The more number of nodes can be added without affecting the service.

Response Time: can be defined as the amount of time taken to react by a load balancing algorithm in a distributed system. For better performance, this parameter should be reduced.

Fault Tolerance: In spite of the node failure, the ability of the system to perform uniform load balancing. The load balancing is the best fault-tolerant technique.

Point of Failure: designed the system in such a way that the single point failure does not affect the provisioning of services. Like in centralized system, if one central node is fail, then the whole system would fail, so load balancing system must be designed in order to overcome this problem.

V. BASIC TYPES OF LOAD BALANCING ALGORITHMS

There is a extremely large need for load balancing in complex and large distributed systems. Load balancer takes a decision to transfer the job to the remote server for load balancing. Load balancer can works in two ways: one is cooperative and non-cooperative. In cooperative way, to achieve the optimal response time, all the nodes work to gather. In non-cooperative way, response time is increase by the independently running the tasks. Some of the algorithms for load balancing are studied in this paper.

Based on the current state of the system, load balancing algorithms can be classified into two types:

- Static algorithm: The current status of the node is not taken into consideration [4]. All the nodes and their properties are known in advance. Based on this prior knowledge, the algorithm works. Since it does not use current system status information, it is easy to implement.

- Dynamic algorithm: This type of algorithm is based on the current status of the system [4]. The algorithm works according to the dynamic changes in the state of nodes. Status Table maintains the Current status of all the nodes in
the cloud. Dynamic algorithms are complex to implement but it balances the load in effective manner.

Based on the initiator of the algorithm, Load Balancing algorithms can be classified into three types [3]:

- **Sender Initiated**: Sender identifies that the nodes are in large number so that the sender initiates the execution of Load Balancing algorithm.
- **Receiver Initiated**: The requirement of Load balancing situation can be identified by the receiver/server in cloud and that server initiates the execution of Load Balancing algorithm.
- **Symmetric**: It is the combination of both the sender initiated and receiver initiated types.

**VI. LOAD BALANCING ALGORITHMS**

Following load balancing algorithms are currently prevalent in clouds

6.1 **Round-Robin Algorithm** [5]

It is the static load balancing algorithm which uses the round robin scheme for allocating job. It selects the first node randomly and then, allocates jobs to all other nodes in a round robin fashion. Without any sort of priority the tasks are assigned to the processors in circular order. Because of the non uniform distribution of workload, this algorithm is not suitable for cloud computing. Some nodes get heavily loaded and some nodes get lightly loaded because the running time of any process is not known in advance. This limitation is overcome in the weighted round-robin algorithm. In the weighted round-robin algorithm some specific weight is assigned to the node. On the basis of assignment of weight to the node it would receive appropriate number of requests. If there are equal assignment of weight, each node receive some traffic. This algorithm is not preferred because prior prediction of execution time is not possible.

6.2 **Opportunistic Load Balancing Algorithm** [6]

This is static load balancing algorithm so it does not consider the current workload of the VM. It attempts to keep each node busy. This algorithm deals quickly with the unexecuted tasks in random order to the currently available node. Each task is assigned to the node randomly. It provides load balance schedule without good results. The task will process in slow in manner because it does not calculate the current execution time of the node.

6.3 **Min-Min Load Balancing Algorithm** [7]

The cloud manager identifies the execution and completion time of the unassigned tasks waiting in a queue. This is static load balancing algorithm so the parameters related to the job are known in advance. In this type of algorithm the cloud manager first deals with the jobs having minimum execution time by assigning them to the processors according to the capability of complete the job in specified completion time. The jobs having maximum execution time has to wait for the unspecified period of time. Until all the tasks are assigned in the processor, the assigned tasks are updated in the processors and the task is removed from the waiting queue. This algorithm performs better when the numbers of jobs having small execution time is more then the jobs having large execution time. The main drawback of the algorithm is that it can lead to starvation.

6.4 **Max-Min Load Balancing Algorithm** [7]

Max Min algorithm works same as the Min-Min algorithm except the following: after finding out the minimum execution time, the cloud manager deals with tasks having maximum execution time. The assigned task is removed from the list of the tasks that are to be assigned to the processor and the execution time for all other tasks is updated on that processor. Because of its static approach the requirements are known in advance then the algorithm performed well. An enhanced version of max min algorithm was proposed in [8]. It is based on the cases, where meta-tasks contain homogeneous tasks of their completion and execution time, improvement in the efficiency of the algorithm is achieved by increasing the opportunity of concurrent execution of tasks on resources.

6.5 **The two phase scheduling load balancing algorithm** [9]

It is the combination of OLB (Opportunistic Load Balancing) and LBMM (Load Balance Min-Min) Scheduling algorithms to utilize better execution efficiency and maintain the load balancing of the system.OLB scheduling algorithm keeps every node in working state of algorithm the cloud manager first deals with the currently available node. Each task is assigned to the node thereby minimizing the overall completion time. This algorithm works to enhance the utilization of resources and enhances the work efficiency.

6.6 **ANT COLONY OPTIMIZATION BASED LOAD BALANCING ALGORITHM** [3]

Aim of the ant colony optimization to search an optimal path between the source of food and colony of ant on the basis of their behavior. This approach aims efficient distribution of work load among the node. When request is initialized the ant starts movement towards the source of food from the head node. Regional Load Balancing Node (RLBN) is chosen in Cloud Computing Service Provider (CCSP) as a head node. Ants keep records the every node they visits ant record their data for future decision making. Ant deposits the pheromones during their movement for other ants to
select next node. The intensity of pheromones can vary on the bases of certain factors like distance of food, quality of food etc. When the job gets successful the pheromones is updated. Each ant build their own individual result set and it is later on built into a complete solution. The ant continuously updates a single result set rather than updating their own result set. By the ant pheromones trials, The solution set is continuously updated.

6.7 Honeybee Foraging load balancing Algorithm [11]

It is a nature inspired decentralized load balancing technique which helps to achieve load balancing across heterogeneous virtual machine of cloud computing environment through local server action and maximize the throughput. The current workload of the VM is calculated then it decides the VM states whether it is over loaded. If the current load of VM is under loaded or balanced, according to the current load of VM they are grouped. The priority of the task is taken into consideration after removed from the overload VM which are waiting for the VM. Then the task is schedule to the lightly loaded VM. The earlier removed task are helpful for the finding the lightly loaded VM. These tasks are known as scout bee in the next step. Honey Bee Behavior inspired Load Balancing technique reduces the response time of VM and also reduces the waiting time of task.

6.8 Biased Random Sampling load balancing Algorithm[12]

Biased Random Sampling Load Balancing Algorithm is dynamic approach, the network is represented in the form of virtual graph. Each server is taken as a vertex of the node and the in degree represents the available free resources the nodes have. On the basis of the in degree the load balancer allocates the job to the node. The nodes have at least one in degree then load balancer allocates the job to that node. When the job is allocated to the node then the in degree is decrement by one, and it’s get incremented again when job gets executed. Random sampling technique is used in the addition and deletion of the processes. The processes are centralized by the threshold value, which indicates the maximum traversal from one node to destination node. The length of traversal is known as walk length. The neighbor node of the current node is selected for the traversal. After receiving the request, load balancer selects a node randomly and compares the current walk length with the threshold value. If the current walk length is equal to or greater than the threshold value, the job is executed at that node. Otherwise, the walk length of the job is incremented and another neighbor node is selected randomly. The performance is decrease as the number of servers increase.

6.9 Active Clustering load balancing Algorithm [13]

Active Clustering is works on the basis of grouping similar nodes and increase the performance of the algorithm the process of grouping is based on the concept of match maker node. Match maker node forms connection between its neighbors which is like is the initial node. Then the matchmaker node disconnects the connection between itself and the initial node. The above set of processes is repeating again and again. The performance of the system is increases on the basis of high availability of resources, because of that, the throughput is also increasing. This increase in throughput is because of the efficient utilization of resources.

7 COMPARISONS OF ALGORITHMS

<table>
<thead>
<tr>
<th></th>
<th>Round Robin</th>
<th>DLB</th>
<th>Min Max 2 phase</th>
<th>Min Max Ant colony</th>
<th>Honey Bee</th>
<th>Biased Random Sampling</th>
<th>Active Clustering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Throughput</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Overhead</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Fault tolerance</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Response Time</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Resource Utilization</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
</tr>
<tr>
<td>Scalability</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Performance</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

8 CONCLUSIONS

Cloud computing provides everything to the user as a service over network. The major issues of cloud computing is Load Balancing. Overloading of a system may lead to poor performance which can make the technology unsuccessful, for the efficient utilization of resources, the efficient load balancing algorithm is required. In this paper, we have surveyed various load balancing algorithms in the Cloud environment. We have discussed the already proposed algorithms by various researchers. The various load balancing algorithms are also being compared here on the basis of different types of parameter.

References


