International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

Research Article

A Comparative Study of Different Static and Dynamic Load Balancing Algorithm in Cloud Computing with Special Emphasis on Time Factor

Abhijit Aditya†*, Uddalak Chatterjee† and Snehasis Gupta†

†Department of Computer Science, WBUT (West Bengal), Burwan, West Bengal, India

Accepted 31 May 2015, Available online 03 June 2015, **Vol.5, No.3 (June 2015)**

Abstract

Cloud computing is distributed technologies to satisfy a variety of applications and user needs. There are various technical challenges that needs to be addressed like Virtual machine migration, server consolidation, fault tolerance, high availability and scalability but central issue is the load balancing. Proper load balancing aids in implementing fail-over, enabling scalability, over-provisioning, minimizing resource consumption and avoiding bottlenecks etc. Various parameters are also identified, and these are used to compare the existing techniques. We present the performance analysis of various load balancing algorithms based on different parameters, considering two load balancing approaches static and dynamic. The analysis indicates that static and dynamic both types of algorithm have some advantages as well as disadvantages. The main purpose is analyze different algorithms based on time factor.

Keywords: *Cloud Computing, Load Balancing, Virtualization, static load balancing, dynamic load balancing.*

Introduction

It is a type of computing in which resources are shared rather than owning personal devises or local personal servers which can be used to handle applications on system. The word cloud in cloud computing is used as a metaphor for internet so we can define a cloud computing as the internet based computing in which the different services like storage, servers and application are provided to organizations computers and device using internet. So as compared to traditional "own and use" technique if we use cloud computing, the purchasing and maintenance cost of infrastructure is eliminated. It allows the users to use resources according to the arrival of their needs in real time. Thus, we can say that cloud computing enables the user to have convenient and on-demand access of shared pool of computing resource such as storage, network, application and services, etc.. on pay per use basis.

Cloud computing provide infrastructure, platform, and software as services. These services are using pay-as-you-use model to customers, regardless of their location. Cloud computing is a cost effective model for provisioning services and it makes IT management easier and more responsive to the changing needs of the business . The access to the infrastructure incurs payments in real currency in cloud environment. In such a model, users access services based on their requirements without regard to where the services are hosted. This model has been referred to as *utility computing*, or recently as *Cloud computing*. The latter term denotes the infrastructure as a "Cloud" from which businesses and users are able to access application services from anywhere in the world on demand. Hence, Cloud computing can be classified as a new paradigm for the dynamic provisioning of computing services, typically supported by state-of-the-art data centers containing ensembles of networked Virtual Machines.

Cloud computing has become a key technology for online allotment of computing resources and online storage of user's data in a lower cost, where computing resources are available all the time, over the internet with pay per use concept. Cloud computing is business oriented concept where computing resources are outsourced by cloud provider to their client, who demand computing online. There is various advantage of cloud computing including virtual computing environment, on-demand services, maximize resource utilization and easy to use services etc. But there are also some critical issues like security, privacy, load management and fault tolerance etc.

Theoretical Background

Cloud Virtualization

In context of cloud computing the virtualization is very worthwhile concept. Virtualization is like "something

*Corresponding author: **Abhijit Aditya**

that is not real" but provides all the facilities that are of real world. This is a software implementation of computer on which different programs can be executed as in the real machine. Virtualization is a part of cloud computing, because different services of cloud can be used by user. All these different services are provided to end user by remote datacenters with full virtualization or partial virtualization manner. There are two types of virtualization which are available and it is described below (Ratan Mishra *et al*, 2012).

- Full Virtualization

In full virtualization (Ratan Mishra *et al*, 2012)the entire installation of one system is done on other system. Due to this all the software that are present in actual server will also available in virtual system and also sharing of computer system among multiple users and emulating hardware located on different systems are possible.

- Para Virtualization

In this type of virtualization (Ratan Mishra *et al*, 2012), multiple operating systems are allowed to run on a single system by using system resources like memory and the processor (VMware software). Here complete services are not fully available, but partial services are provided. Disaster recovery, migration and capacity management are some salient features of Para virtualization.

Cloud computing is Internet based computing, whereby shared resources, software and information are provided to computers and other devices on-demand, like a public utility.

Cloud Infrastructure

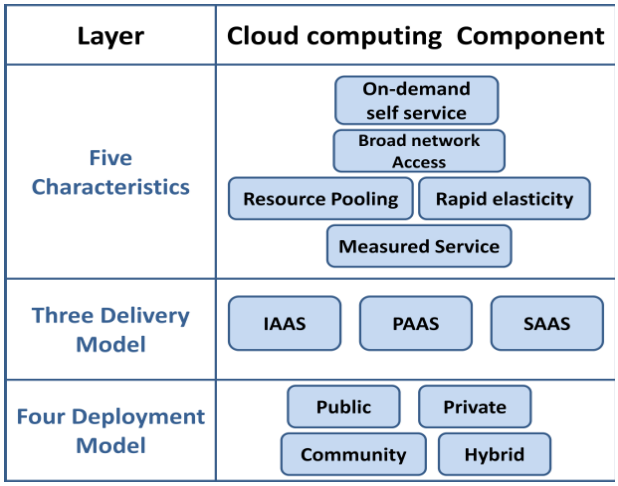

Fig.1: Cloud Infrastructure (Tushar Desai *et al*, 2013)

Characteristics

Cloud computing is an on-demand, virtualized, cost-effective, elastic, location and device independent, and all time available system. Cost is reduced to a significant level as the infrastructure is provided by a third-party and global utilization of resources which avoids wastage of resources and computing power. The main goal of cloud computing is to make a better use of distributed resources by combine them to achieve higher throughput and be able to solve large scale computation problems. Some of the most important key characteristics are:

On-demand Self Service

Computing resources are provided online according to the client requirement at specific time without any human interaction. In cloud computing, users access the data, applications or any other services with the help of a browser only, regardless of the other software and hardware.

Broad Network Access

Cloud users have broad area of cloud services those are accessible through internet. There is no dependency of client platform to access cloud services. Services are always on, anywhere, anytime and anyplace.

Resource Pooling

The cloud computing resources are pooled to serve multiple consumers according to consumer demand.

Measured Service

Cloud users do not need to control and optimize computing resource because all are automatically managed by cloud system. Resource usage can be monitored, controlled, and reported for providing transparency for both the provider and consumer of the utilized service.

Selection of Provider

Selection of cloud service provider is the key to get good service. So according to their choice and their knowledge of cloud provider, users can select the right service provider. One must make sure that the provider is reliable and well-reputed for their customer service and also should have a proven track record in IT-related ventures.

The types of cloud computing technology can be viewed from two perspectives: *Capability* and *Accessibility* (Carnegie Mellon *et al*, 2010):

Cloud Delivery Model

There are three cloud delivery models, Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS)

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): Cloud computing deliver a SAAS service where user do not need to manage installation and configuration of any hardware or software. All the installation and configuration of services are managed by vendor or cloud provider. *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".

Main advantages are reduced up-front cost, potential for reduced lifetime cost, elimination of licensing risk, elimination of version compatibility and reduced hardware footprint etc.

The main disadvantage of SAAS is billing management, synchronization of client and vendor migrations etc.

Example of SaaS provider is *Google Apps,* Google Online office, Google Docs, Email cloud etc. that provides large suite of web based applications for enterprise use.

Platform-as-a-Service (Paas): It is a delivery of a computing platform over the web where user can create and install their own application as their need. Configuration of computing platform and server is managed by vendor or cloud provider. Web applications can be created quickly without complexity of buying and managing the storage server, database, and other software/hardware.

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.
Advantage: PASS Enables developers to focus on application code, Instant global platform Elimination of H/W dependencies and capacity concerns Inherent scalability.

Disadvantages are Strong governance required to prevent lines of business from building applications without IT involvement.

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): Infrastructure As A Service

Infrastructure of servers, software, and network equipment is provided as an on-demand service by the cloud provider. 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. *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: Cloud Deployment Model

There are four cloud deployment models as public, private, hybrid and community clouds.

Public Cloud

Public cloud allows users to access the cloud publicly via interfaces using web browsers. Users need to pay only for the time duration they use the service, i.e., pay-per-use: 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. This can be compared to the electricity system which we receive at our homes. We pay only for the amount of that we use. The same concept applies here. This helps in reducing the operation costs on IT expenditure. However public clouds are less secure compared to other cloud models as all the applications and data on the public cloud are more prone to malicious attacks. The solution to this can be that security checks be implemented.

Private Cloud

A private clouds operation is within an organization's internal enterprise data center. 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. The main advantage here is that it is easier to manage security, maintenance and upgrades and also provides more control over the deployment and use. Private cloud can be compared to intranet. Compared to public cloud where all the resources and applications were managed by the service provider, in private cloud these services are pooled together and made available for the users at the organizational level.

Hybrid Cloud

It is a combination of public cloud and private cloud. In this model a private cloud is linked to one or more external cloud services. It is more secure way to control data and applications and allows the party to access information over the internet. It enables the organization to serve its needs in the private cloud and if some occasional need occurs it asks the public cloud for intensive computing resources.

Community Cloud

When many organizations jointly construct and share a cloud infrastructure, their requirements and policies then such a cloud model is called as a community

cloud. The cloud infrastructure could be hosted by a third-party provider or within one of the organizations in the community.

Load Balancing in Cloud Computing

In computing, the load may be of various types like memory load, CPU load and network load etc. Load balancing is the process of searching overloaded node and transferring the extra load of the overloaded node to other nodes which are under loaded, for improving resource utilization and decreasing server response time of the jobs. When the size of cloud scales up, cloud computing is required to handle massive data accessing requests, such as distributed data mining. Load balancing is the process of reassigning the total load to the individual nodes of the collective system to make effective resource utilization and to improve the response time of the jobs, simultaneously removing a condition in which some of the nodes are over loaded while some others are under loaded. A key challenge on those applications is that clouds have to keep the same or better performance when an outburst of data accessing request occurs, i.e., In GIS application, an area becomes hot search area when a disaster takes place, and heterogeneous nodes with different computing are unbalanced.

The important things in said load balancing are estimation of load, comparison of load, stability of different system, performance of system, interaction between the nodes, nature of work to be transferred, selecting of nodes and many other ones to consider while developing such algorithm. To improve the performance substantially, backup plan in case the system fails even partially, maintain the system stability, accommodate future modification in the system are main goal of load balancing.

Load balancing in the cloud computing based on standard load balancing but differs from classical thinking on load-balancing architecture and implementation by using commodity servers to perform the load balancing, which provides for new opportunities and economies of scale as well as presenting its own unique set of challenges. The load balancers served to promote availability of cloud resources and to promote performance. In complex and large systems, there is a need for load balancing to simplify it in a cloud computing environment.

Basic Types of Load Balancing Algorithms

There is an 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 (Venubabu Kunamneni,2012). 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(Venubabu Kunamneni,2012). 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 (Ratan Mishra *et al*, 2012) :

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.

Following static load balancing algorithms are currently prevalent in clouds

1. *Round Robin Algorithm(Pooja Samal, Pranati Mishra,2013)*

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. The Round Robin algorithm ((Pooja Samal, Pranati Mishra,2013) mainly focuses on distributing the load equally to all the nodes. Using this algorithm, the scheduler allocates one VM to a node in a cyclic manner. The round robin scheduling in the cloud is very similar to the round robin scheduling used in the process scheduling. The scheduler starts with a node and moves on to the next node, after a VM is assigned to that node. This is repeated until all the nodes have been allocated at least one VM and then the scheduler returns to the first node again. Hence, in this case, the scheduler does not wait for the exhaustion of the resources of a node before moving on to the next. 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.

Advantage

- The main advantage of this algorithm is that it utilizes all the resources in a balanced order.
- An equal number of VMs are allocated to all the nodes which ensure fairness.

Disadvantage

- In this method it considers current load on each virtual machine.
- 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.

2. Opportunistic Load Balancing Algorithm (OLB) (Che-Lun Hung1 et al, 2012)

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.

Advantage

- 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.

Disadvantage

- 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.

3. Min-Min Algorithm (T. Kokilavani et al ,2011)

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 unspecific 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.

It begins with a set of all unassigned tasks. First of all, minimum completion time for all tasks is found. Then among these minimum times the minimum value is selected which is the minimum time among all the tasks on any resources. Then according to that minimum time, the task is scheduled on the corresponding machine. Then the execution time for all other tasks is updated on that machine by adding the execution time of the assigned task to the execution times of other tasks on that machine and assigned task is removed from the list of the tasks that are to be assigned to the machines. Then again the same procedure is followed until all the tasks are assigned on the resources. But this approach has a major drawback that it can lead to starvation.

Advantage

- This algorithm performs betters when the numbers of jobs having small execution time is more then the jobs having large execution time.

Disadvantage

- The main drawback of the algorithm is that it can lead to starvation.

4. Max-Min Load Balancing Algorithm (T. Kokilavani et al, 2011)

Max-Min is almost same as the min-min algorithm except the following: after finding out minimum execution times, the maximum value is selected which is the maximum time among all the tasks on any resources. Then according to that maximum time, the task is scheduled on the corresponding machine. Then the execution time for all other tasks is updated on that machine by adding the execution time of the assigned task to the execution times of other tasks on that machine and assigned task is removed from the list of the tasks that are to be assigned to the machines. 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. 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.

Advantage

Max-min strategy resolves the difficulty of Min-min, by giving

Priority to large tasks. The Max-min algorithm selects the task with the

Maximum completion time and assigns it to the resource on which achieves minimum execution time. It is clear the Max-min seems better choice whenever the number of small tasks is much more than large ones.

Disadvantage

One of the features of the Max-min strategy is that chooses large tasks to be

Executed firstly, which in turn small task delays for long time.

5. The two phase scheduling load balancing algorithm (OLB+LBMM)

OLB (Opportunistic Load Balancing) and LBMM (Load Balance Min-Min) (Karanpreet Kaur *et al* 2013) scheduling algorithms to utilize better executing efficiency and maintain the load balancing of the system. This combined approach helps in an efficient utilization of resources and enhances the work efficiency. It gives the better results than the above discussed algorithms. 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 to achieve the goal of load balance and LBMM scheduling algorithm is utilized to minimize the execution of time of each task on the node thereby minimizing the overall completion time. This algorithm works to enhance the utilization of resources and enhances the work efficiency.

This algorithm performs the following steps for load balancing (Karanpreet Kaur *et al* 2013):

- Average completion time of each task for all the nodes is calculated.
- Select the task with maximum average completion time.
- Select an unassigned node with minimum completion time that should be less than the maximum average completion time for the selected task. Then, the tasks dispatched to the selected node for computation.
- If all the nodes are already assigned, re-evaluate by considering both assigned and unassigned nodes. Minimum completion time is computed as:
- Minimum completion time of the assigned node is the sum of the minimum completion time for all the tasks assigned to that node and minimum completion time of the current task.
- Minimum completion time of the assigned node is the minimum completion time of the current task.
- Repeat step 2 to step 4, until all tasks are executed. This gives better results than the above discussed algorithms.

Advantage

- Efficient utilization of resources.
- Enhances work efficiency.

Disadvantage

- No fault tolerance.

Various parameters have been considered to compare different static algorithm.

- The metrics on which the existing load balancing techniques have been measured are discussed below:
  1. Throughput
- This metric is used to estimate the total number of tasks, whose execution has been completed successfully. High throughput is necessary for overall system performance.
  2. Overhead
- Overhead associated with any load balancing algorithm indicates the extra cost involved in implementing the algorithm. Overhead Associated determines the amount of overhead involved while implementing a load-balancing algorithm. It includes overhead due to movement of tasks, inter-processor and inter-process communication. It should be as low as possible.
  3. Fault Tolerance
- It measures the capability of an algorithm to perform uniform load balancing in case of any failure.
- A good load balancing algorithm must be highly fault tolerable.
  4. Migration Time
- It is defined as, the total time required in migrating the jobs or resources from one node to another.
- It should be minimized.
  5. Response Time
- It can be measured as, the time interval between sending a request and receiving its response. It should be minimized to boost the overall performance.
  6. Resource Utilization
- It is used to ensure the proper utilization of all those resources, which comprised the whole system.
- This factor must be optimized to have an efficient load balancing algorithm.

7. Scalability

- It is the ability of an algorithm to perform uniform load balancing in a system with the increase in the number of nodes, according to the requirements. Algorithm with higher scalability is preferred.

8. Performance

- It is used to check, how efficient the system is. This has to be improved at a reasonable cost, e.g., reducing the response time though keeping the acceptable delays.

Comparison of existing static load balancing techniques based on different performance parameters with special reference to time factors

Table 1: Various metrics have been considered to compare different techniques

Comparative study on different factors							Time analysis	
Metrics/ Techniques	Throughput	Overhead	Fault tolerance	Resource Utilization	Scalability	Performance	Migration Time	Response Time
Round robin	Yes	Yes	No	Yes	Yes	Yes	No	Yes
Olb+lbmm	No	No	No	Yes	No	Yes	No	No
Min-min	Yes	Yes	No	Yes	No	Yes	No	Yes
Max-min	Yes	Yes	No	Yes	No	Yes	No	Yes
Olb	No	No	No	Yes	No	Yes	No	No

Description on basis of Time analysis for StaticLoad Balancing Algorithm

- *Round Robin Algorithm:* Round Robin Algorithm deals in static and decentralized cloud system. Over here, Migration time is low, but there is Response time that means the total time required in migrating the jobs or resources from one node to another is low but the time interval between sending a request and receiving its response is little high which should be minimized boost the overall performance. 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.

- *Opportunistic Load Balancing Algorithm:* This is static load balancing algorithm so it does not consider the current workload of the VM. It attempts to keep each node busy. Migration time and Response time is low, that means the total time required in migrating the jobs or resources from one node to another is low and the time interval between sending a request and receiving its response is also low which boost the overall performance. This algorithm deals quickly with the unexecuted tasks in random order to the currently available node. It attempts to keep each node busy. But the task will process in slow in manner because it does not calculate the current execution time of the node.

- *Min-Min algorithm:* 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 unspecific period of time. Over here, Migration time is low, but there is Response time that means the total time required in migrating the jobs or resources from one node to another is low but the time interval between sending a request and receiving its response is little high which should be minimized boost the overall performance. The job with the smallest execution time is executed. Some jobs may experience starvation.

- *Max-Min Load Balancing Algorithm:* Max-Min being a static approach the requirements are known in advance then the algorithm performed well. In Max-Min algorithm after finding out minimum execution times, the maximum value is selected which is the maximum time among all the tasks on any resources. Then according to that maximum time, the task is scheduled on the corresponding machine. Over here, Migration time is low, but there is Response time that means the total time required in migrating the jobs or resources from one node to another is low but the time interval between sending a request and receiving its response is little high which should be minimized boost the overall performance. Execution time for all other tasks is updated on that machine by adding the execution time of the assigned task to the execution times of other tasks on that machine and assigned task is removed from the list of the tasks that are to be assigned to the machines.

- *The two phase scheduling load balancing algorithm (OLB+LBMM):* 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 to achieve the goal of load balance and LBMM scheduling algorithm is utilized to minimize the execution of time of each task on the node thereby minimizing the overall completion time. This algorithm works to enhance the utilization of resources and enhances the work efficiency.

Migration time and Response time is low, that means the total time required in migrating the jobs or resources from one node to another is low and the time interval between sending a request and receiving its response is also low which boost the overall performance.

Dynamic Load Balancing Algorithm

This type of algorithm is based on the current status of the system. 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. In dynamic load balancing algorithms work load is distributed among the processors at runtime. The master assigns new processes to the slaves based on the new information collected. Unlike static algorithms, dynamic algorithms allocate processes dynamically when one of the processors becomes under loaded. Instead, they are buffered in the queue on the main host and allocated dynamically upon requests from remote hosts.

Dynamic load balancers continually monitor the load on all the processors, and when the load imbalance reaches some predefined level, the redistribution of work takes place. But as this monitoring steals CPU cycles so care must be taken as when it should be invoked. This redistribution does incur extra overhead at execution time.

Dynamic Load Balancing (DLB) Algorithms

1. ACCLB (ant colony and complex network theory) (Ratan Mishra et al, 2012): It is a load balancing mechanism based on ant colony and complex network theory in an open cloud computing federation. 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.

Advantage

- It uses small-world and scale-free characteristics of a complex network to achieve better load balancing. This technique overcomes heterogeneity, is adaptive to dynamic environments, is excellent in fault tolerance and has good scalability hence helps in improving the performance of the system.
- Excellent in fault tolerance
- Good scalability

Disadvantage

- Throughput is less.

2. Honeybee Foraging Algorithm (Dhinesh B. L.D et al, 2013): The main idea behind the Honeybee Foraging algorithm is derived from the behavior of honeybees. There are two kinds of honeybees: finders and reapers. The finder honeybees first goes outside of the honey comb and find the honey sources. After finding the source, they return to the honey comb and do a waggle dance indicating the quality and quantity of honey available. Then, reapers go outside and reap the honey from those sources. After collecting, they return to beehive and does a waggle dance. This dance indicates how much food is left. M. Randles proposed a decentralized honeybee based algorithm for self-organization. In this case, the servers are grouped as virtual server and each virtual server have a process queue. Each server, after processing a request from its queue, calculates the profit which is analogous to the quality that the bees show in their waggle dance. If profit is high, the server stays else, it returns to the forage. This algorithm requires that each node to maintain a separate queue. This computation of profit on each node causes additional overhead. 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 ,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.

Advantage

- Achieves global load balancing through local serve actions.
- Performs well as system diversity increases.

Disadvantage

- The disadvantage of this algorithm is that, it does not show any significant improvement in throughput, which is due to the additional queue and the computation overhead.

3. Biased Random Sampling load balancing Algorithm (Randles M et al, 2010): 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 allocates 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 neighbour 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 neighbour node is selected randomly. Whenever a client sends a request to the load balancer, the load balancer allocates the job to the node which has at least one in-degree. Once a job is allocated to the node, the in-degree of that node is decremented by one. After the job is completed, the node creates an incoming edge and increments the in-degree by one. The addition and deletion of processes is done by the process of random sampling. Each process is characterized by a parameter know as threshold value, which indicates the maximum walk length. A walk is defined as the traversal from one node to another until the destination is found. At each step on the walk, the neighbour node of current node is selected as the next node. In this algorithm, upon receiving the request by the load balancer, it would select 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. Else, the walk length of the job is incremented and another neighbour node is selected randomly.

Advantage

- Achieves load balancing across all system nodes using random sampling of the system domain.
- Performs better with high and similar population of resources.

Disadvantage

- The performance is degraded as the number of servers increase due to additional overhead for computing the walk length. Degrades as population diversity increases.

4. Active Clustering load balancing Algorithm (Ram Prasad Padhy et al,2011): 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 neighbours which is like as 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. Active Clustering is a clustering based algorithm which introduces the concept of clustering in cloud computing. The performance of an algorithm can be enhanced by making a cluster of nodes. Each cluster can be assumed as a group. The principle behind active clustering is to group similar nodes together and then work on these groups.

The process of creating a cluster revolves around the concept of match maker node. In this process, first node selects a neighbor node called the matchmaker node which is of a different type. This matchmaker node makes connection with its neighbor which is of same type as the initial node. Finally the matchmaker node gets detached. This process is followed iteratively.

Advantage

- The performance of the system is enhanced with high availability of resources, thereby increasing the throughput. This increase in throughput is due to the efficient utilization of resources.

Disadvantage

- Degrades as system diversity increases.

Comparison of Existing Dynamic Load Balancing Techniques based on Different Performance Parameters with Special Reference to Time Factors

Table 2: Various metrics have been considered to compare different techniques

Comparison of different factors							Time analysis	
Metrics/ Techniques	Throughput	Overhead	Fault tolerance	Resource Utilization	Scalability	Performance	Migration Time	Response Time
Honeybee foraging	No	No	No	Yes	No	Yes	No	No
Bias random sampling	Yes	Yes	No	Yes	No	Yes	No	No
Active clustering	Yes	Yes	No	Yes	No	No	Yes	No
ACCLB	No	No	Yes	Yes	Yes	Yes	No	No

Description on basis of Time analysis for Dynamic Load Balancing Algorithm

1. ACCLB (Ant Colony and Complex Network Theory): (ACCLB) in an open cloud computing federation. It uses small-world and scale-free characteristics of a complex network to achieve better load balancing. Migration time and Response time is low, that means the total time required in migrating the jobs or resources from one node to another is low and the time interval between sending a request and receiving its response is also low which boost the overall performance. This technique overcomes heterogeneity, is adaptive to dynamic environments, is excellent in fault tolerance and has good scalability hence helps in improving the performance of the system.

2. Honeybee Foraging Algorithm: Honeybee Foraging Algorithm is a nature-inspired algorithm for self-organization. It achieves global load balancing through local server actions. Migration time and Response time is low which boost the overall performance. Performance of the system is enhanced with increased system diversity but throughput is not increased with an increase in system size. It is best suited for the conditions where the diverse population of service types is required.

3. Biased Random Sampling load balancing Algorithm: Biased Random Sampling Load Balancing Algorithm is dynamic approach, over here Migration time and Response time is low which boost the overall performance. It uses random sampling of the system domain to achieve self-organization thus balancing the load across all nodes of the system. The performance of the system is improved with high and similar population of resources thus resulting in an in-creased throughput by effectively utilizing the increased system resources. It is degraded with an increase in population diversity.

4. *Active Clustering load balancing Algorithm:* Active Clustering load balancing Algorithm is a self-aggregation algorithm to optimize job assignments by connecting similar services using local re-wiring. The performance of the system is enhanced with high resources thereby in-creasing the throughput by using these resources effectively. But in this algorithm the Migration time is high so the performance degraded with an increase in system diversity.

Conclusion

Load balancing is one of the main challenges in cloud computing. It is required to distribute the dynamic local workload evenly across all the nodes to achieve a high user satisfaction and re-source utilization ratio by making sure that every computing re-source is distributed efficiently and fairly. With proper load balancing, resource consumption can be kept to a minimum which will further reduce energy consumption and carbon emission rate which is a dire need of cloud computing. Existing load balancing techniques that have been discussed mainly focus on reducing associated overhead, service response time and improving performance etc. 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. The purpose of this paper was to compare different load balancing algorithms based on identified qualitative parameters. In this paper we have carried out the analysis of different load balancing algorithms, various parameters are used to check the results. Load balancing algorithms is totally dependent upon in which situations workload is assigned, during compile time or execution time. The above comparison shows that static load balancing algorithms are more stable than dynamic. But dynamic load balancing algorithms are always better than static as per as overload rejection, reliability, adaptability, cooperativeness, fault tolerant, resource utilization, response & waiting time and throughput is concert. In future work, we need more and more real experimentation to choose good load balancing algorithm.

Reference

Ratan Mishra and Anant Jaiswal(2012), Ant Colony Optimization: A solution of Load Balancing in Cloud, *International Journal of Web & Semantic Technology (IJWesT)*.

Tushar Desai, Jignesh Prajapati(2013) A Survey Of Various Load Balancing Techniques And Challenges In Cloud Computing, *International Journal Of Scientific & Technology Research* , Volume 2, Issue 11.

Carnegie Mellon, Grace Lewis (2010), Basics About Cloud Computing, Software Engineering Institute

Venubabu Kunamneni (2012),Dynamic Load Balancing for the cloud , *International Journal of Computer Science and Electrical Engineering*

Pooja Samal, Pranati Mishra, (2013) ,Analysis of variants in Round Robin Algorithms for load balancing in Cloud Computing , *International Journal of Computer Science and Information Technologies*, Vol. 4 (3) , 416-419

Che-Lun Hung1, Hsiao-hsi Wang2 and Yu-Chen Hu2(2012), Efficient Load Balancing Algorithm for Cloud Computing Network, *IEEE* ,Vol. 9, pp: 70-78

T. Kokilavani, Dr. D. I. George Amalarethinam (2011),Load Balanced Min-Min Algorithm for Static Meta Task Scheduling in Grid computing, *International Journal of Computer Applications* ,Vol-20 No.2

Karanpreet Kaur, Ashima Narang, Kuldeep Kaur (2013),Load Balancing Techniques of Cloud Computing, *International Journal of Mathematics and Computer Research*

Dhinesh B. L.D , P. V. Krishna (2013), Honey bee behavior inspired load balancing of tasks in cloud computing environments, *Applied Soft Computing*, volume 13, Issue 5, Pages 2292-2303

Randles M., Lamb D. and Taleb-Bendiab A. (2010) 24*th International Conference on Advanced Information Networking and Applications Workshops*, 551-556.

Ram Prasad Padhy ,P Goutam Prasad Rao,(2011), Load Balancing in Cloud Computing Systems, National Institute of Technology, Rourkela, India.