An Efficient Load Balancing based on Resource Utilization in Cloud Computing

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Software Engineering

Submitted By
Gurpreet Singh Bedi
(Roll No. 801231009)

Under the supervision of:
Ms. Ashima Singh
Assistant Professor,
Computer Science and Engineering Department

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR UNIVERSITY
PATIALA – 147004

June 2014

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, *"An Efficient Load Balancing based on Resource Utilization in Cloud Computing"*, in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Software Engineering* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Ms. Ashima Singh* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Gurpreet Singh Bedi

(Gurpreet Singh Bedi)

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Ashima Singh 4/6/14.

(Ms. Ashima Singh)
Assistant Professor,
Computer Science and Engineering Department
Thapar University

Countersigned by

(Dr. Deepak Garg)
Head
Computer Science and Engineering Department
Thapar University
Patiala

(Dr. S. K. Mohapatra)
Dean (Academic Affairs)
Thapar University
Patiala

ABSTRACT

The development of the internet has given birth to many technologies. Cloud computing is a latest trend in large scale data processing. It aids in providing shared resources. It gives support to the distributed parallel processing. However, providing an efficient response time is a major challenging issue in the distributed systems. Cloud computing provides data on the pay per use basis and eliminates the need of having one's own device. As cloud computing grows, more users get attracted towards it. Lesser response time is needed for distributed computing and effective load balancing is one of the major issues that can improve response time. Improving the dynamic nature of load balancing algorithms in order to improve the performance of the cluster is the first and foremost requirement. In the proposed algorithm, load balancing is done by considering priority policy. Priority calculation is done by considering hardware parameters including CPU speed, memory resource and power consumption which helps to avoid the overloading and underloading of resources. A resource allocation strategy that takes into account resource utilization would lead to better energy efficiency. An Efficient Load Balancing based on Resource Utilization is proposed and related algorithm is executed on cloudsim and its toolkit. The results show the effectiveness of the proposed algorithm.

ACKNOWLEDGEMENT

First of all I would like to thank the Almighty, who has always guided me to work on the right path of the life. This work would not have been possible without the encouragement and able guidance of my supervisor *Ms. Ashima Singh*. I thank my supervisor for her time, patience, discussions and valuable comments. Her enthusiasm and optimism made this experience both rewarding and enjoyable.

I am also thankful to the entire faculty and staff members of Computer Science Department, Thapar University, for their direct-indirect help, cooperation, love and affection.

Last but not the least, I would like to thank my parents for their wonderful love and encouragement, without their blessings none of this would have been possible. I would also like to thank my close friends for their constant support.

LIST OF CONTENTS PAGE NO.

LIST OF FIGURES PAGE NO.

LIST OF TABLES PAGE NO.

Chapter 1

INTRODUCTION

1.1. Introduction

Cloud computing is the ability of using various computing resources through the internet including applications and storage services. The shared pool of the resources is hosted by the cloud provider. According to the National Institute of Standard and Technology [1], Cloud Computing is defined as a model for providing convenient and on demand access to the shared pool of resources including networks, storage, services etc. These services require minimal effort. From the business point, cloud computing is a new business model that has not been possible without the cloud. The main attribute of the cloud computing is the elasticity, means its tendency to grow and reduce the computation as per the requirements [2]. The other attribute is it is scalable, means its ability to balance the increased demands of the CPU storage, bandwidth etc. Cloud Computing provides the secure access to the applications as shown in the Figure 1.1. But high level security is a challenge for the cloud developers.

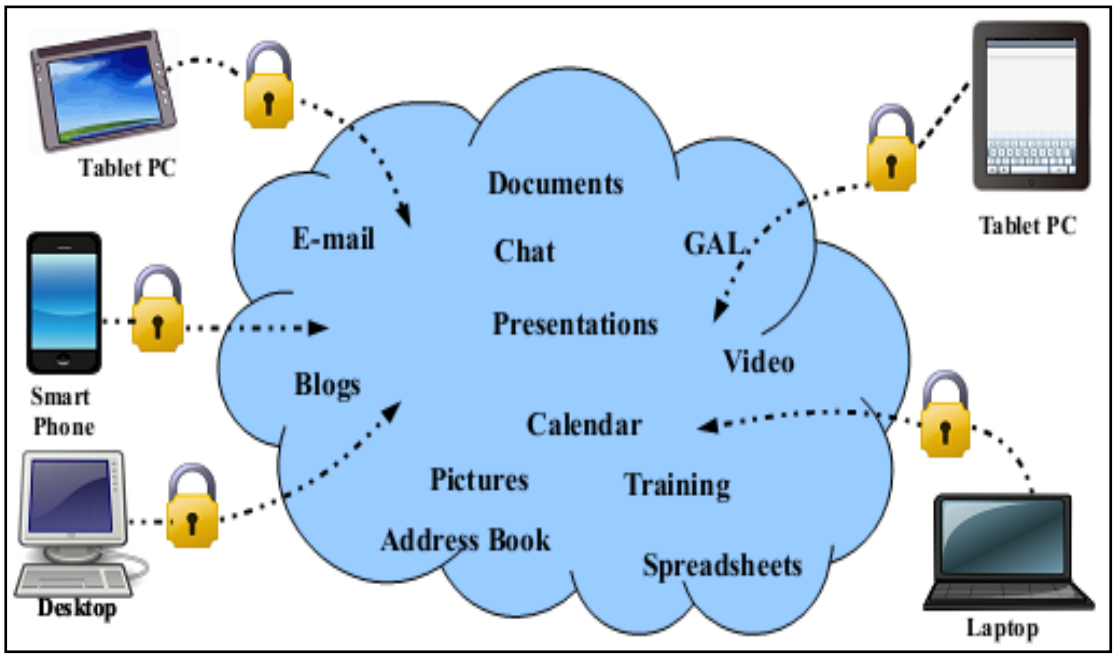

Figure 1.1: Cloud Computing [3]

The traditional computing resources are scalable but are not elastic. The resources in the cloud computing can be used by more than one tenant. Also the services are metered means one can pay only for the resources it consumes whereas in traditional systems the price is fixed based on the requirements.

The term “cloud” was originated in 1990’s, when for the data communication Virtual Private Networks (VPN) was used to provide services [4]. VPN’s enabled dynamic routing which increased the efficiency of the bandwidth and thus helped to balance the network utilization. Thus the term “telecom cloud” was proposed. However concept of the cloud computing was introduced by John McCarthy in 1960 and he suggested to have computation- a public utility in the future. Prior to the VPN, companies provided the dedicated point to point service which resulted in the waste of the bandwidth and thus resulted in the increase of the overall cost. It is a recent trend in the IT sector. To meet the user needs, Cloud Computing provides the virtual environment. It is based on Service Oriented Architecture (SOA) to provide hardware support and it moves the data, processing and service delivery away from the desktop transform to the data centers [5]. It provides computation, storage access and other services that may not require the end user knowledge of configuration of the system and the physical location. The main aim of the cloud computing is to make use of the distributed resources to increase the overall throughput. It is a model which provides on demand data access to the shared pool of the resources [6]. Cloud computing is similar to the electricity as the electric components are attached to the central grid rather than the production of the electricity by their own. This is beneficial in terms of cost and the time. Cloud is distributed into many levels- client, server, application, infrastructure and the platform. Current computing (Cloud) is similar to the old computing (Mainframe Computers) but the difference lies in speed, storage size, memory and the cost. Cloud computing is far better than the mainframe computing but direct access to the cloud resources increases the threat to the security [7]. This is the only main hurdle in the cloud computing. Researchers are working to solve these issues and providing the solutions with the help of the frameworks and strategies. For example vulnerability is the potential weakness which opens the door to the attacker is the main concern.

1.1.1. Advantages of Cloud Computing

I. Ease in management- A person needs only internet connection and the web browser to access the services without the need for installing any other application for the computation. The maintenance of the infrastructure requires less time and the cost. The applications can also make easier access to the storage services through the cloud.

II. Reduction in the cost- It reduces the cost because costly infrastructure does not require here- Most of the applications deployed on the cloud does not need any man power and can be setup freely like emails, Google Apps. Also these applications are reliable and the main advantage of such applications is availability.

III. No interruption in the services- Cloud computing provides uninterrupted services to the user [6].

IV. Disaster Management- Cloud computing is also efficient in the disaster recovery. Keeping the backup of an important data is need of the organizations. An offsite backup is helpful to recover the data. Cloud storage not only keep the back up of the data offsite but also ensures that they have system provided for disaster recovery in terms of the failure.

V. Green Computing- Energy consumption is the main concern in the today scenario including electronic waste with the advancement in time, extensive use of the system resources. This can be reduced with the help of the cloud computing to some extent. Less e-waste generated results in environment preserving.

VI. Easy to scale- The cloud resources are managed by the software and thus if new requirement arises then it is easy to scale up the cloud. The scalability process can be done in less time. Similarly the resources can also scale down with the requirements.

1.2. Cloud Computing Infrastructure

Cloud computing architecture is divided into two parts- the front end and the back end. The front end means the User Interface (UI), the client sees and it is responsible for taking the request from the user as an input. The back end means the cloud system which is responsible for the whole computations. Both front end and the back end are connected via the network .To access the cloud front end has the computer as the front

end. The central server is responsible for monitoring the traffic and act as an admin to the system. The networked computers are communicated by the middleware.

1.2.1. Layers of Cloud Computing Architecture

There are different layers in the cloud computing. First layer is the cloud client consists of the hardware or the software, relies on the cloud to provide the services. Second layer is the application layer which provides Software as a service (SaaS), which eliminates the need to install an application on the user's computer. It allows the customers to access the services via the internet. Access and management both are done from the centralized locations remotely.

Software as a Service (SaaS)	Platform as a Service (PaaS)	Infrastructure as a Service (IaaS)
1. Communication (Emails etc.) 2. Productivity Tools(Office)	1. Application Development 2. Database Management 3. Security Services	1. Management 2. Network 3. Storage 4. Servers
Example – Oracle, IBM, Google Apps, etc.	Example- Amazon EC2, Microsoft Azure, etc.	Example- GoGrid, Flexiscale etc.

Table 1: Overview of the organizations which make use of cloud as a service

The organizations such as Google Apps, Microsoft, Oracle are the key providers of providing Software as a Service (SaaS) as shown in Table 1. Third layer provides platform as a service (PaaS) which uses cloud infrastructure. Through this one can get all the facilities including developing, deploying, hosting and testing of the applications. So the users need not to install the software and hardware. All the applications which are required by the client are distributed across the network. For example Microsoft Azure is the provider of the infrastructure as the service. Fourth layer provides Infrastructure as the service (IaaS). Here the main advantage is that, the customer need to pay only for the time they spend to use the infrastructure. The client needs not to purchase any server or resources. This also enables fast access to the data with the less cost. Most relevant examples include GoGrid and Flexiscale. Last layer

is the server layer which includes the hardware/software helps to provide the above services. So Cloud computing is the good option for the organizations to reduce the cost and increase the computational speed. Figure 1.2 shows an overview of layers which use Cloud Computing to provide the services. As shown in the Figure 1.2, in IaaS, applications, security and databases are customer managed layer, whereas the other layers are provider managed. In PaaS, only the application layer is customer managaged whereas in SaaS, all the layers are provider managed.

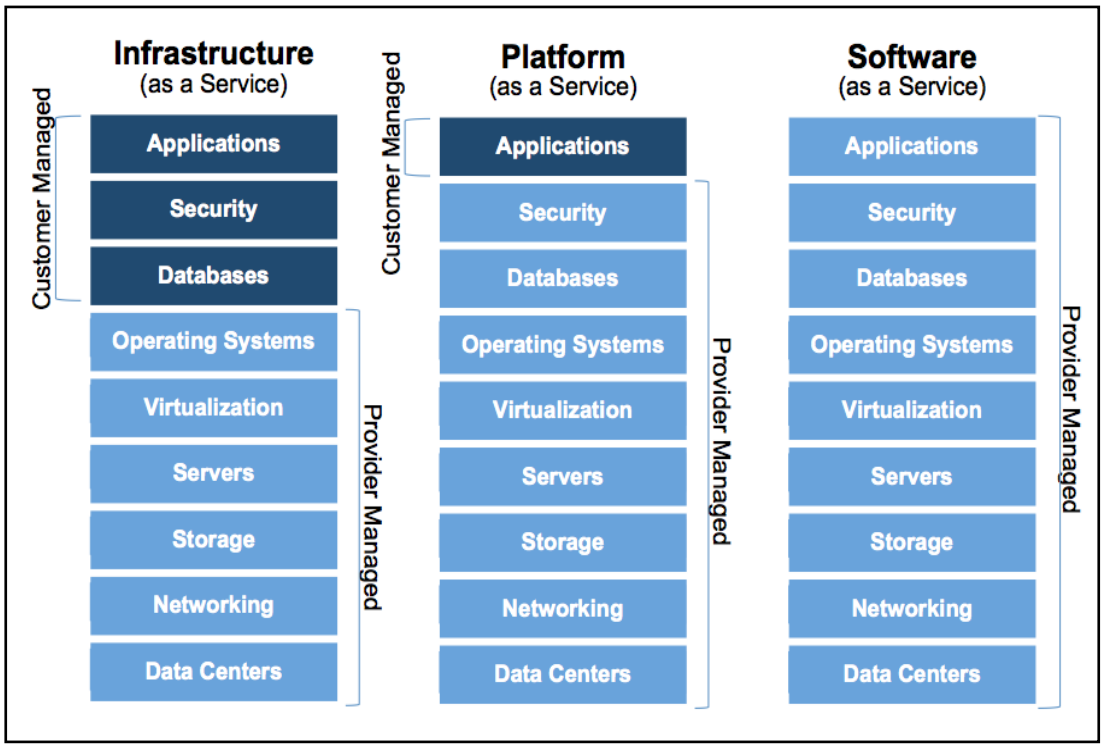

Figure 1.2: Comparison between three main layers of Cloud Computing [8]

Today, the demand of exchanging of the information on the network is continuously increasing. Cloud computing offers IT services to the users globally. Its architecture is mainly based on five layers including Client Layer, Application Layer, Platform Layer, Infrastructure Layer and Server layer with three levels of abstraction is later discussed in the thesis. It is based on pay per use model. The five layer architecture has shown in Figure 1.3. Cloud computing has many advantages including reduction in the cost of the technology infrastructure, globalized workflow, improvement in the flexibility etc. However there are some issues too. For example- security concern, prone to attack, resource utilization performance, load balancing in the distributed system etc.

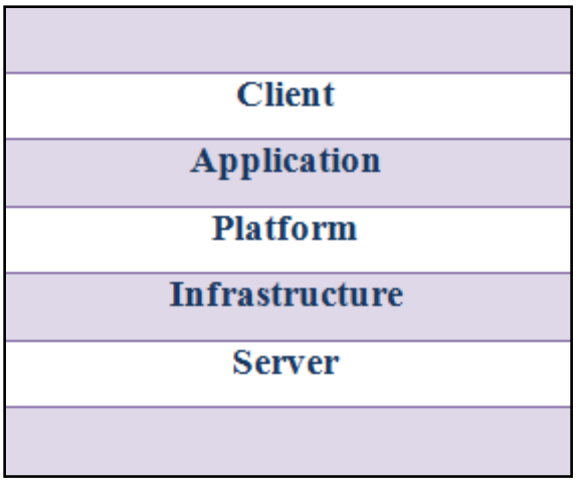

Figure 1.3: Five Layers of Cloud computing

1.2.2. Cloud Stakeholders

Cloud Computing has different stakeholders and various issues which are covered as shown in Table 2. These stakeholders as shown below-

I. End User- An end user can access cloud infrastructure. The main of cloud infrastructure to minimize the cost and it is available on-demand basis. The main challenge is the security of the cloud. Private clouds are more secure as compared to the public and the hybrid clouds. An end user can use the services and can pay as per the usage. Cloud provides the flexible approach to the services. Quality of Service parameters is checked by the users before using the cloud services. A SLA (Service Level Agreement) is signed to meet the quality standards.

II. Cloud Provider- cloud provider is responsible for building the cloud. Based on the requirements scenarios there are different types of clouds including private, public and hybrid cloud are discussed in the thesis. The provider can offer any of the clouds.

III. Cloud Developer- Developer meets technical requirements and covers the technical details of the cloud computing. The attributes or the issues which are handled by the cloud developer have shown in the table. The developer is there to bridge the gap exist between the two i.e. the provider and the end user. The developer lies between the two i.e. Cloud Provider and the Cloud End User.

Types of Stakeholders	Issues
End User	• Ease of Use • Security • Privacy • Reduced Cost • Availability
Cloud Provider	• Energy Efficiency • Outsourcing • Resource Utilization • Meet Requirements • Managing Resources
Cloud Developer	• Virtualization • Availability • Reliability • Data Management • Programmability

Table 2: Cloud Stakeholders

1.2.3. Cloud Deployment

The main concern is the type of the cloud to be deployed. There are basically four types of the clouds to be deployed. A brief architecture has shown in Figure 1.4.

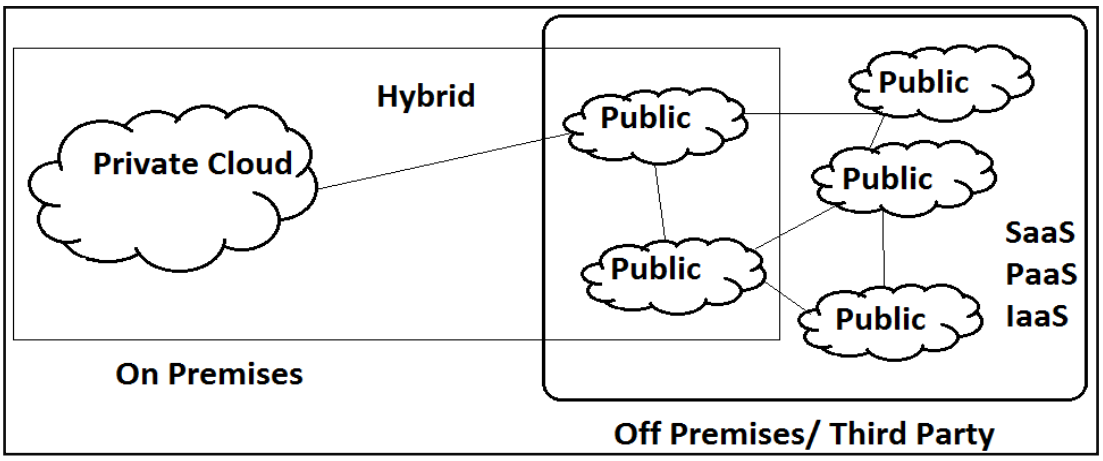

Figure 1.4: Cloud Deployment

I. Public cloud- As stated earlier in this thesis, public cloud means allowing access to the cloud via the interfaces including the web browser. This is the pay per use service similar to the electricity we consume and we pay in accordance with the usage. However public clouds are less secure and are more prone to the attacks.

This issue can be solved by providing the security on both the cloud aswell as the client. The implemented security checks reduce the threat of the security.

II. Private cloud- In this all the operations are done within the organization. The main advantage of using the private cloud in comparison with the public cloud is that, in private cloud maintenance and deployment is easier. Here resources are pooled together and provided at the organization level so less threat to the security. Security is more enhanced and easily managed. For example intranet in the organizations provides the private cloud.

III. Hybrid cloud- A hybrid cloud is mainly used to serve the needs in the private cloud and also if requires, it facilitates the services in public cloud too as shown in figure 2. So a hybrid cloud may be the mixture of public or private cloud linked with each other through the network. It is a good way to provide the security for the required services over the internet.

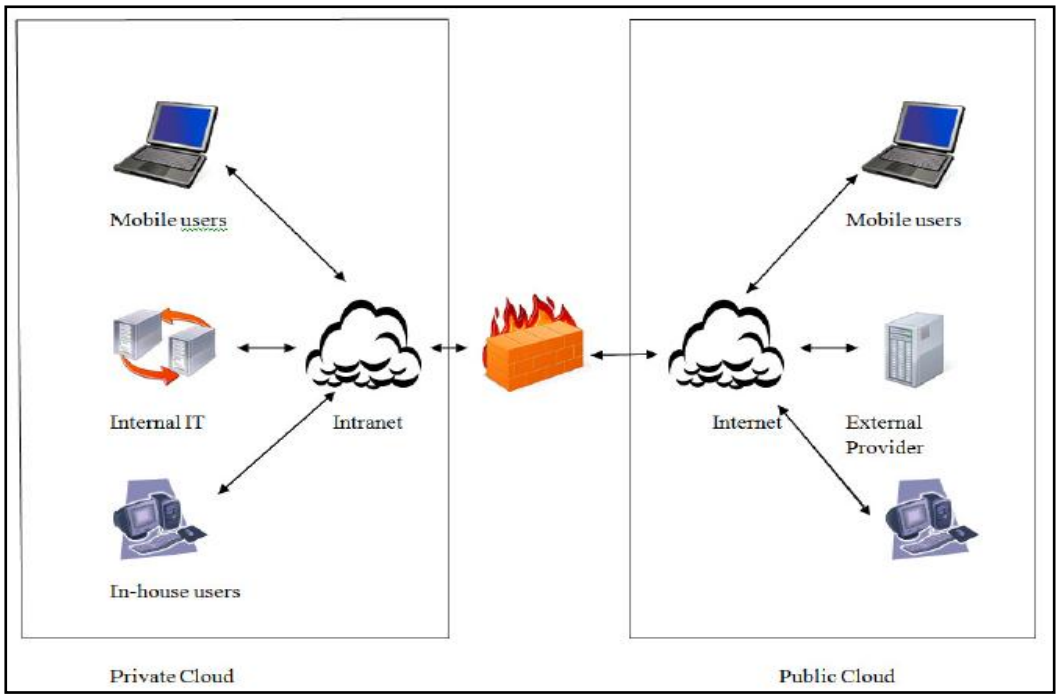

Figure 1.5: An overview of Hybrid Cloud [9]

IV. Community cloud- The community cloud may be provided by the third party or within the organization. A cloud is called community cloud if many organizations share their policies and the requirements in the cloud. So a community cloud is constructed by the many organizations and access to the cloud is done remotely.

1.3. Cloud Computing comparison with Cluster and Grid Computing

Cluster is the base of the parallel and the distributed computing. It combines the resources locally and thus shares the workload. It takes place under the supervision of only one administrative domain. The integration of the nodes is responsible for the whole computation. Grid is the extended version of the cluster computing, as the resources are shared through the internet geographically. It is the combination of the resources which are distributed on the network. The integration of the nodes permits sharing of the resources dynamically. Cloud computing is similar to the grid computing to some extent as both cloud and grid computing is for parallel distributed systems. But cloud computing is not a single domain. It is the collection of the various domains which provide various services. Nodes in the cloud computing are virtualized and it provides real time computation. Also the resources in the grid computing is pre reserved, whereas the resources in the cloud computing are demand driven, means resources are provided at real time according with the customer needs [9]. So cloud is more flexible then the grid computing and it provides on-demand resources. Resource management in the grid computing is always distributed. However in the cloud computing, resource management may either be distributed or centralized.

1.4. Characteristics of Cloud Computing

I. User can access the data and perform the computations without the help of any device. Cost gets reduced as the infrastructure which provided by the third party now can be accessed via the internet and the maintenance is also easy as it doesn't require the physical setup.

II. Performance can be monitored and measured by pay per use facility. It also requires fewer skills for the implementation.

III. Sharing of the resources provides efficient utilization of the bandwidth [10].

IV. The privacy is the main concern in the cloud computing, as the data shared on the cloud may be confidential. But it provides better security then traditional systems because providers are able to distribute the resources and solve the security issues. The customers might not afford to solve themselves these issues.

V. People are connecting with the cloud and thus getting many benefits including emails, application software, instant messaging and that too with the less cost.

VI. It provides the virtual environment on the basis of the three models- public, private and the hybrid. Public cloud means the infrastructure is maintained and owned by the service provider. In private, the model is owned and maintained by the organization and the hybrid cloud may be the composition of two or more clouds that could be public or private [11].

VII. Cloud Computing helps to delivers the services that were not possible before [12]. For example- Mobile based applications provide real time data processing through cloud are location aware and context aware. Weather monitoring applications are also managed through clouds. Batch processing enables to process the entire data (size in terabytes) parallel. Apache Hadoop Mapreduce makes the entire complex process easier through batch processing. Business analysts use the large volume of the data to understand the customers. Cloud computing also reduces the latency in the bandwidth.

1.5. Virtualization

Virtualization in cloud computing is the method of providing the resources. For example hardware resources as virtual machines. Virtual machine (VM) is known as the guest. A host is responsible for providing the virtual environment. Virtualization gives the illusion of the real thing but it is not real. It provides all the features of the real thing. An end user can use all the services of the virtualized thing just as the real object. So virtualization is related with the cloud to provide the services to the end user. The datacenter is able to provide the services in two ways either in full virtualization or in Para virtualization [13].

In the full virtualization, full installation of the machine is done. After the installation of one machine on the other, the new machine contains all the software of the previous machine. Full virtualization able to delivers the services among the multiple users. It also isolates the users from one other. So a successful sharing is done among the multiple users.

Para virtualization is responsible for the care of efficient use of the memory and the resources. It allows multiple OS to run on the single machine. In this approach, the services are available partially. The main advantage of using this approach including disaster recovery, capacity management and the migration. In case of the system

failure, instances of the system are moved to the other machine and the failure is recovered. Migration of the instances is simple and easy. Hence the power management is also an easy task in the case of the Para virtualization.

1.5.1. Resource allocation

Allocation is to provide the resources on the demand basis. The main aim is to keep the track of the overloaded node so that no wastage of the resources. The wastage implies the wastage of the CPU speed, memory or the bandwidth. The entire mapping is done in the two levels-

The first level is the mapping of the virtual machine to the host. Virtual machine resides on the physical servers known as the hosts. A VM is mapped to the host and the process depends on the capacity and the availability. Allocation depends on the on-demand requirements.

The second level is Application mapping to the virtual machine. For execution, an application requires some power. Virtual machines provide the power as the applications are executed on the virtual machines. Applications are mapped to Virtual machines and this is dependent on the availability and the configuration.

1.5.2. Task Scheduling

The scheduling is done after the allocation of the relevant resources. Scheduling specifies the manner in which the resources are allocated. Allocation means which resources are available to meet the requirements and the scheduling specifies in which way the allocation is to be done. It checks if the resources are available individually or on the shared basis. It basically provides the feature of multiprogramming. Scheduling is done in two ways either as a space shared or as on the time shared. Both the virtual machine and the host are allocated in either way. The main difference in both the modes is that the resources are preempted in the case of the time sharing mode. In case of the space sharing mode, resources are not preempted.

There are four cases to be considered:

Case1: Both the virtual machine and the hosts are allocated in the time shared mode.

Case2: Only the virtual machine is allocated in the time shared mode but the hosts are allocated in the space shared mode.

Case3: Only the virtual machine is allocated in the space shared mode but the hosts are allocated in the time shared mode.

Case4: Both the virtual machine and the hosts are allocated in the space shared mode.

1.6. Load Balancing

A web server has the dispatcher to balance the incoming request to the servers. The main aim of dispatcher in load balancing is to transfer the request to the server that is available at that time. The front end is responsible to balance the requests by making decisions regarding the transfer so that the load is transferred efficiently to the server which can process the request at that moment. Web server's load information is used in making the decisions by the front end. The front end may send a series of the requests to the number of the web servers. An example of the load balancing has shown in the Figure 1.6. In this, the front end distributes the load to the server with least load at that moment. The server and the front end exchanges the information about the load with each other to make the effective decisions. With the help of the correct decision about the load balancing, the service quality is improved and the system becomes more robust. The load balancer decides how to forward the requests and the decision is made correspond to the CPU load percentage on the particular virtual machine.

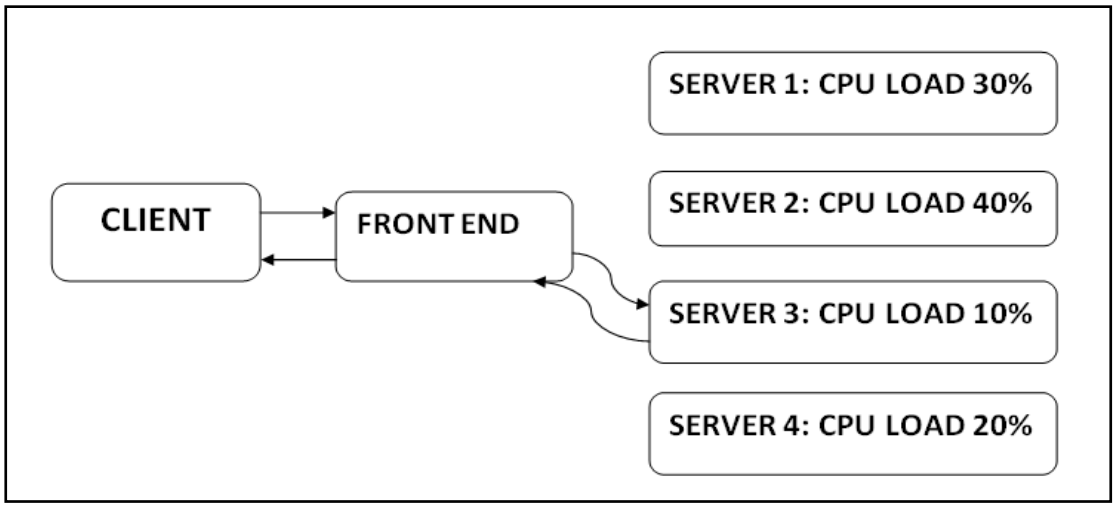

Figure 1.6: Load Balancing

1.7. Motivation

Cloud Computing is a vast research. The internet is viewed as the cloud which provides either connection less or connection oriented services. After studying many research thesis, I have found several issues in cloud computing. But the main focus is on load balancing. It is one of the main challenges in cloud computing. To combat with the issues, many load balancing techniques has been proposed. The main concern is to maximize the throughput. The main focus in this thesis is to increase the performance of the system by the proper utilization of the virtual machines. So a new load balancing technique is proposed.

1.8. Barriers in Cloud Computing

There are many benefits in using cloud computing technology. However there are some barriers too as shown below-

I. Security- Security is the main concern at all levels including network, host and the client.
II. Access and Connectivity- Cloud computing depends on the availability of the high speed network access. The sharing of the information through the cloud is another factor that aids in industrialization but it depends on the network access [14].
III. Reliability- The applications must be available 24x7 to provide services without interruption. In case of the service failure, backup plans must be there and must begin without any ill effect. However to ensure the reliability additional costs may be added to mitigate risks.
IV. Interoperability- For the adoption of the cloud computing, interoperability between private and public clouds is another concern. Now depending on the task, it may require additional effort to integrate with the traditional applications that may be situated on another cloud. So the maintenance of the interoperability is the another barrier in the cloud computing.
V. Economic Cost- Cloud computing is cost effective means of data sharing on the cloud. However there may be hidden cost associated with the process which includes- data recovery, application modification, application insurance etc. For example it is not cost effective to use multiple Software as a service (SaaS) to solve one task. The interoperability in cloud computing may

includes the risk of increase in cost in transition of the solution from one cloud to another.

VI. Changes in IT sector- There is a major shift in technology in IT sector. The main challenge is the adopting the new skill to solve the problem.

VII. Global Issues and boundaries- There is variability in the cloud computing. The location of the physical data, processing and access of the data takes place at different locations. So political barrier is there in the adoption of cloud computing as certain rules and regulations may apply in adoption of the technology. This results in negative impact on cloud computing. For example countries like Canada are making law –USA Patriot Act to ensure the privacy and security of the data [15]. Canada told the organizations not to use the computers globally which are within US borders. However Amazon Web services have solved this issue to some extent by making Amazon Virtual Private Cloud.

1.9. Objectives of Study

I. Detailed study of the load balancing techniques and their comparison.
II. Comparison and analysis of Load Balancing Algorithms.
III. Proposal of Efficient Load Balancing Algorithm.
IV. Implementation and Analysis of Results.

1.10. Organization of Thesis

The thesis has been divided into 6 chapters as listed below:

Chapter 1: "Introduction to Cloud Computing" gives an overview on Cloud Computing, characteristics of Cloud Computing, its barriers and various issues.

Chapter 2: "Literature Survey of Load Balancing" describes the various Load Balancing Techniques and establishes the problem based upon the literature survey.

Chapter 3: "Problem Statement" describes the issues in the previous approaches.

Chapter 4: "Problem Formulation" describes the proposed Load Balancing Technique.

Chapter 5: "Experiment Results" provides the analysis of the proposed solution.

Chapter 6: "Conclusion and Future Scope" provides the conclusion of the proposed solution and the possible future extension.

Chapter 2

LITERATURE SURVEY

Load Balancing is a technique to distribute the workload across various nodes, or other resources like central processing units, network links, etc. The main aim is to achieve efficient resource utilization, increase in the throughput, decrease the response time and minimize the overhead. It also helps to achieve the fairly distribution of the workload.

A DNS server is scheduled to translate the address to the particular IP when accessing the particular web services through the addresses. This URL translation is tasked to select a particular node from the cluster. This is based on the scheduling policy of the web servers. A period is defined to cache all the translations in terms of TTL (Time To Live). After the time to cache the translation is expired, the next task is routed to the server. Round Robin is the way to implement the load in the rotating way. Hence the load balancing is achieved in DNS servers.

In the network based approach, software/hardware is installed as a front end to balance the load to the based on the data found in the protocols including the network layer protocol or the data layer protocol. This is done by the process of routing. Apache can also be used as a HTTP load balancer which fetches the requests from the servers and after processing delivers the requests to the users. It also keeps tracks of the sessions that allow a single user to deal with the single server.

Cloud computing provides a model in which a number of the resources can be shared by on-demand service. To share the resources effectively, load balancing plays a vital role, which decides how the request can be processed. The decision effects the sharing of the resources in a multiple server cluster. A dispatcher is there which is responsible for receiving the request and distributes the workload to the back end servers. The dispatcher can be implemented in many ways. For example Network Address Translation (NAT), IP tunneling etc. In NAT, both the incoming and the outgoing data is passed through the dispatcher whereas in IP tunneling or the direct routing, the data can be inserted through the dispatcher but the outgoing data can be directly passed to the client/user. This approach is not popular due to the security concerns.

The load balancing is achieved by the relevant hardware/software, for example multilayer switch helps in load balancing. It is also one main issue of cloud computing as there may occur a situation in which some nodes are busy and at the same time some nodes are idle. So this increases the overall response time. A mechanism is needed to ensure that the workload across the nodes is evenly distributed. Different algorithms are there to serve the efficient load balancing.

2.1. Load Balancing Scenarios Based on the Nature of Nodes

The nature of node in the cloud computing is responsible for load balancing. The algorithm of load balancing depends on the decision made by the node. There are mainly three categories of load balancing including centralized, distributed and hierarchical load balancing.

I. Centralized balancing

In this technique, all the decisions are made by the single node. This node is responsible for managing the entire network. The process may be static or dynamic depends on the requirement specification. This process decreases the response time to analyze the different resources as the whole management is done by the centralized node. There are also some limitations of using this approach. For example this technique has high failure intensity and is no fault tolerant due to an overhead mainly on the centralized node. Also the recovery after the failure is not an easy task in case of centralized load balancing.

II. Distributed balancing

In this technique, no single node is responsible for the entire scheduling decision. There is efficient distribution of the tasks across the multiple domains which are responsible for the entire function. Here no single node is overloaded. Hence the overhead in the distributed load balancing is less as compared to the centralized load balancing. Honeybee foraging and biased random sampling are some examples of the distributed load balancing.

III. Hierarchical balancing

This technique mainly operates in the master slave node. A parent node is responsible to balance the nodes. The architecture of hierarchical load balancing is based on tree

data structure. Master node can use its agents to get the information of the slave nodes. The scheduling is done upon the collection of the information by the parent node.

Nature of the Algorithm	Base Knowledge	Advantages	Limitations
Static	Prior knowledge is required	Usage in homogeneous cluster	If there is change in requirements Not scalable/Flexible
Dynamic	Based on Run time statistics	Usage in heterogeneous cluster	Complex structure More Time consuming
Centralized	Single node is responsible	Usage in small networks	Single node overhead No fault tolerance
Distributed	All the nodes are responsible	Usage in large and heterogeneous cluster	Complexity
Hierarchical	Nodes at different levels of hierarchy	Usage in medium/large heterogeneous cluster	Less fault tolerance
Workflow Dependent	Decision made on dependencies of the graph	Usage in heterogeneous/ homogenous cluster	Maintenance is complex

Table 3: Categories of Load Balancing Algorithms

IV. Dependent tasks

Dependent tasks are those tasks which are dependent on the subtasks. There execution takes place after the execution of the subtasks. The scheduling of the tasks is based on the workflow based algorithms. These algorithms uses directed and acycyclic graph (DAG) as the knowledge base. Workflows are further classified into two types including transaction incentive and data incentive. In transaction incentive workflows, multiple instances of the workflow have the same structure. Data incentive workflows are used when the size of the data is large.

2.1.1. Factors that affect Load Balancing

Response Time- The main factor that affects the performance of the load balancing algorithms is the architecture. In decentralized, centralized and hierarchical load balancer architecture, with the increase in the number of the users, the response time increases. However, hierarchical approach performs better then the centralized and decentralized approach [16]. Therefore, the hierarchical approach takes less time as compared to the other approaches. The centralized and decentralized approach shows similar response time.

Server Load- Server load means the number of the requests the system can handle per second, it can be represented as requests/sec. The aim is to split the workload across the three architectures to check their ability to handle the workload. Both centralized and the decentralized load balancers, the servers show the similar load with the increase in the number of the nodes. The experiment was performed on the three architectures [16] and the results showed that the performance of the hierarchical load balancer is far better due to its architecture and the ability of the load balancing algorithm to maintain the centralized management. Load balancing technique in the cloud computing helps to maintain the distribution of the workload across the nodes and thus reduces the amount of energy consumed. It helps in avoiding the overheating of the cluster. So, the energy efficient cloud contains the four main elements including the consumer, Resource allocator, Virtual machine and the physical machine [17]. These four elements are shown in the Figure 2.1 and are described below-

I. Consumer

Consumer may request a service from anywhere globally. For example- In a company, a web application that runs through the internet can act as a consumer. This application distributes the workload as per the requirement. Requirement is based on the number of the users using the application.

II. Resource Allocator

Resource Allocator act as an interface between the consumer and the cloud infrastructure. It has further many components including green negotiator, service analyzer, energy monitor, energy scheduler, etc. Green negotiator makes the service level agreement between the cloud provider and the user to maintain the quality of

service. Service Analyzer checks the requirements specification and takes decision on whether to accept the request or to reject it. To achieve the load balancing strategy, it needs efficient information from the virtual machine and the energy monitor. Virtual machine manger manages the workload across the virtual machines. Service Scheduler helps to schedule the requests across the virtual machines for effective utilization of the resources.

III. Virtual Machines

Virtual machines (VM) can be started or stopped on the request. Multiple VM can run concurrently and thus provides the flexibility to update the data resources on the single physical machine. By the migration of the virtual machine across the physical machine, the nodes can be put together to save the energy resources.

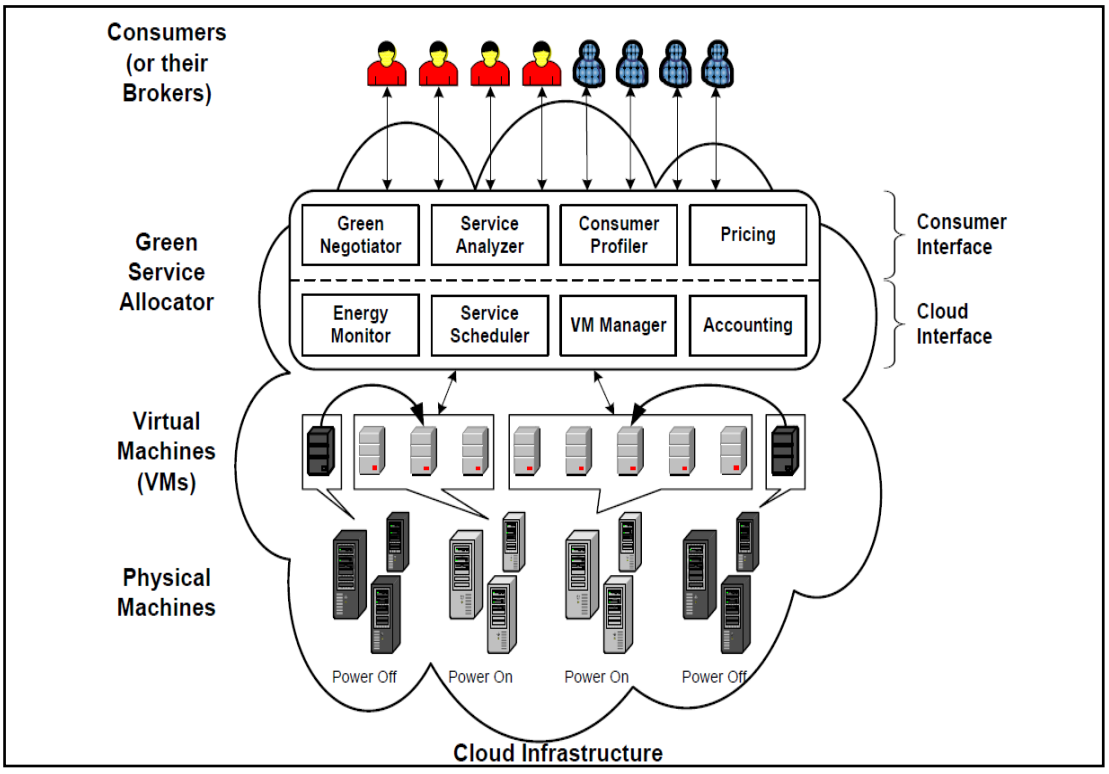

Figure 2.1: Cloud Computing Infrastructure

IV. Physical Machines

Physical machine act as a hardware resource to meet the requirement specification.

2.1.2. Cloud Computing Optimization

There are two major concepts in load balancing- system load and system topology [18]. The first one is system load which depends on how the workload is loaded on

the computer nodes. System load may follow centralized approach, distributed or hybrid approach. In centralized approach, a single node able to manage the whole distribution. In distributed approach, each node is responsible to build its own load vector. This is done by collecting the relevant information by the other nodes. This approach then makes the decision by checking the load vector. In cloud computing, this technique is more efficient in comparison with the centralized approach. Mixed approach is the mixture of the distributed and the centralized approach. The second is the system topology means this approach depends on the study of the information status of the nodes. System topology is further classified into three approaches- static approach, dynamic approach and adaptive approach. Static approach depends either on the design/architecture or on the implementation of the system. Dynamic approach on the other hand depends on the decision making during load balancing to efficient distribute the workload. The next is the adaptive approach which is most suitable if the system state changes most frequently. This approach makes decisions by changing the parameters of the system when the state changes.

In cloud computing, mainly three level of abstraction is there including Infrastructure as a service(IaaS), software as a service(SaaS) and platform as a service(PaaS). The levels of abstraction are shown in the Figure 2.2. In cloud computing, a cloud system has various data centers, which further includes various machines. Each machine then distributes the workload efficiently to various virtual machines with the purpose to save the energy [19]. The actual execution is done on the host machine but the client/user sees the resources on the virtual machine.

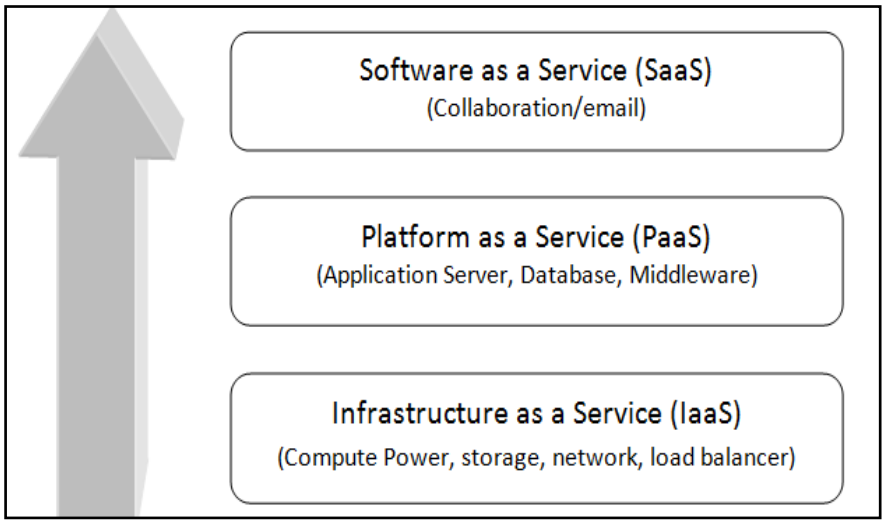

Figure 2.2: Three level abstraction in Cloud Computing

The resource manager in cloud computing can work at two levels- local level and global level. The load balancer at the local level is used to find the host machine of the local data center. The load balancer at the global level is used to find the data center globally. The requirements may change after the resources are allocated to the cloud system. This may lead to the unbalancing situation which needs optimization for better performance. There are several policies which are implemented to increase the throughput. The policies including selection Policy which follows some selection criterion and decides which node needs to be migrated [20], Transfer Policy finds the tasks to be performed at the time of the migration, and Location Policy depends on the selection of the lighter node to find the suitable machine and the Information Policy that collects the information of the nodes.

2.2. Big Data and Cloud

The data is increasing gigantically in many forms like audio, text or video. It is difficult to handle that data with the traditional technologies. Big Data has attracted huge attention from the researchers. Big Data is new emerging technology to handle that data which aids in large scale data processing. MapReduce partitions its work and distributes its work to various nodes/machines [21]. YARN is another version of the mapreduce which provides additional features [22]. But the problem arises when the work gets loaded into the slow running machine, because run time of a machine depends on the various parameters including its processor speed. MapReduce has basically three parts map, shuffle and the reduce part. Map task initiate the mapping of the task. This step partition the task into subtasks. The reduce task groups the similar items. As the map tasks executes in parallel, this can reduce in over loaded of the reduce tasks. Therefore an algorithm was proposed to reduce the overload [23]. The load balancing algorithm in the middle of the map and the reduce tasks, able to divide the large tasks into smaller and then only send the smaller tasks to the reduce tasks and the whole procedure depends on the availability.

2.3. Load Balancing Algorithms

The aim of the load balancing algorithms is to increase the overall performance. The algorithm must provide scalability to ensure timely modification in case of the system state changes.

2.3.1. Static Load Balancing Algorithms

The static load balancing algorithms are based on the prior knowledge of the properties of the nodes and its ability to process the new requests. This also includes the processing power of the node. This only takes into consideration the static properties of the node and is not adaptable to any changes. A central load balancing decision model (CLBDM) was proposed. It is an improvement to the round robin [24]. In this model, a connection time between the user and the node is calculated and is compared against the threshold value. If the connection time exceeds the threshold value then the connection will be terminated. The task will then assign to the next node using the round robin strategy.

In this approach, processes are allocated to the processor in the round robin manner and this allocation is done locally. The main limitation of this approach is that as the processing time of the different processes are not the same so some nodes may remain idle while the other nodes may busy. However this approach is suitable in the web servers where the https requests are distributed equally to handle the web traffic across the network.

2.3.2. Dynamic Load Balancing Algorithms

Dynamic load balancing algorithms depend on the run-time information collected of the selected nodes. In this approach, workload is assigned and may reassign to the nodes based on the computation. These algorithms also require continuous monitoring of the tasks. This is also more accurate approach of load balancing in comparison with the static approach.

2.4. Related Work

Kumar Nishant et al. proposed an algorithm based on Ant's Behavior. Ant's Algorithm is based on the ant's behavior as the ants move towards the region of large amount of resources [25]. An ant is always in search of new food and uses this food sources to deliver back to the nest. In this algorithm, first a head node is chosen. The selection of the head node is based on the node which has large number of the neighboring nodes. The ant is always started from the head node. Like an ant moves in one direction at a time in search for the food and after getting the food it moves

backward towards its nest, similarly this algorithm proposed that the ant moves in forward direction encounter the overloaded or under loaded node. If the ant finds an overloaded node while previously it finds under loaded node, then it will move backward and check if the node is still under loaded. If it is under loaded then the work is distributed across the under loaded nodes. So this is an efficient technique of resource utilization. But the main limitation is that due to large amount of ants the network may be congested. The status of the node after the ant's visit is also not taken into account.

Shu Ching Wang et al. proposed Load Balancing Min-Min (LBMM) algorithm which framework consists of three levels [26]. First is the request manager, second is the service manager and the third one is the subdivision. The request manager is responsible for receiving the workload and then assigns it to the service manager. After receiving the request the service manager further subdivide the workload to increase the processing speed. The service manager may also assign the load to the service node based on several parameters including the CPU remaining memory, node availability etc. However a three level process is slower than the other algorithms as process is to be completed in the three levels. Min-Min algorithm first checks the minimum completion time of the tasks. After the selection of the task with minimum time, the task is assigned to the node and the new time is updated by integrating the previous processing time of the machine with the time of the new task assigned. The assigned task is then removed from the list of the tasks yet to be assigned. The main limitation of using Min-Min algorithm is that it may lead to starvation.

Junjie Ni et al. proposed Load balancing algorithm for virtual machines [27] which contains central scheduling to calculate the resources available for the task. The available resource is then assigned for the processing. Resource monitor is there to calculate the details of the resources available. The algorithm is based on the machine mapping. The process consists of several stages including acceptance of the request, getting the resource information, controller calculates the resource availability to process the tasks. The resource with the highest score is assigned the task. The client will then able to access the application. The limitation of using this algorithm is that it does not take into account the node capabilities and the network load. Also if the single point fails, the whole system gets shut down.

Che-Lun Hung et al. proposed Load Balancing Max-Min-Max Algorithm. It is a two phase algorithm which is a combination of Opportunistic Load Balancing (OLB) and Load Balance Min-Min (LBMM) scheduling algorithms [28]. In OLB load balancing, a task is assigned to the node randomly. It keeps each and every node of the process in the working state. Thus it achieves load balancing. In LBMM technique, information about the nodes is available in the advance, thus it is also known as static algorithm. In LBMM, execution time is reduced. The joint approach of both OLB and LBMM is to maintain load balancing. In the combined job, first the completion time of each job is checked and then the average time of the job is calculated. Second step is to select the job with the maximum response time. Third step is to select the unoccupied node with the minimum response time. The task is then assigned to the selected node. If the nodes are already busy then re compute the result by checking the occupied and un-occupied nodes. Repeat the above steps until all the tasks are executed.

Martin Randles et al. proposed Honeybee Foraging Algorithm [29]. This algorithm is based on the behavior of the honeybees. Two types of honeybees are there- one who finds the honey and the other who reaps. Finder honeybees go in search of the honey and find the food sources. After finding, they return back to their comb and does waggle dance which indicates the quality and the quantity of the honey they find. The reapers then go and reap the honey from the resources which was founded by the finder honeybees. After collecting the honey, reapers honeybees come back to their comb and does waggle dance which indicates the amount of honey left in the sources. Similarly in the honeybee foraging algorithm in cloud computing, several servers are grouped together as a virtual server. Here the servers are like honeybees and the web applications are like food sources. A forager chooses the random virtual server. For processing the request, each server/node maintains a queue. After the processing, the server computes the total profit. Depends upon the profit contribution, server processes the request. If the calculated profit is less, the server returns to their forage. So this maintains the balance of the load of the system. However the computation of the profit may cause an additional overhead which results in overall decrease in the throughput.

Biased Random Sampling balances the load of the system and is based on the dynamic and random sampling of the nodes. In this technique, load of the system is

determined by the virtual graph constructed to represents the connected nodes. Representation of the node is constructed by the vertex and free resources in the system are determined by the in-degree. So allocation of the work depends on the in-degree of the node to balance the workload. When the task is allocated, the in degree is decremented by the value one, which means there is a decrease in the availability of the free resource. After the completion of the request, the node created an edge and the value of the in-degree is incremented by the value one which means that there is an increase in the availability of the free resource. This whole process is done by random sampling [30]. The process is computed by the comparing parameter known as the threshold parameter. Threshold value is represented as maximum walk length. Walk length is determined by calculating the traversing from one node to another until a destination is reached. After receiving the request, the random node is selected by the load balancer, and the current node walk length is compared with the threshold level. If the walk length is greater than that of the threshold value, the task is assigned to that node. If the walk length of the node is less than the threshold value, the job is assigned to the next node and the current walk length is incremented by one. Next node is represented as the neighbor of the current node. This algorithm thus balances the workflow, but the computation of the walk length creates an additional overhead.

Active clustering [31] is a method to balance the load in cloud computing. The main aim of active clustering approach is to group together the similar nodes. The grouped node act as a cluster. The algorithm then works on these groups. The creation of the cluster depends on the match maker node. In this technique, a node selects the neighbor node of the different type known as the matchmaker node. The matchmaker node then makes connection with the neighbor node of the similar type as the initial node. At last the matchmaker node is detached. This algorithm follows iterative approach.

Yi Lua et al. proposed Join-Idle-Queue algorithm [32]. This algorithm is used for large scale load balancing of the distributed data. This technique load balances the idle processors. It mainly concerns the availability of the idle processors across the dispatchers. The main idea is to decouple the discovery of lightly loaded processors from the task. The main advantage of this technique is to reduce the average queue length. The other advantages include less communication overhead and reduction in

the system load. But the main limitation of this technique is that it is not scalable. So it can't be used for today web services which are dynamic in nature. Join idle queue depends on two level system's behavior. First is dispatcher behavior- Dispatcher first receives a job from the client. It checks if there are servers in the queue, if yes then the job is allocated to it else it chooses the random server. This is known as the primary load balancing. Second is the server's behavior means after the completion of the jobs, the balancing of the idle processors across the dispatchers.

Jeffrey M. Galloway et al. proposed Power Aware Load Balancing (PALB) Algorithm [33]. PALB algorithm was designed with the purpose to give computation control to the cluster controller. PALB has three main sections including balancing section, upscale section and the downscale section. Balancing section checks the state of the virtual machine. If all the active nodes computed are with more than 75% utilization, then the load is distributed to the new virtual machine with the least load, else the load is distributed to the most utilized node with the purpose to balance the workload. The threshold level of 75% is chosen. Upscale section is responsible to power up the additional nodes if the current utilization of the nodes is more than 75%. Downscale section is used to power down the underutilized nodes to save the power cost. So PALB gives the shutdown signal to the idle node. Shutdown signal is given to the nodes with less than 25% utilization. This algorithm mainly checks and decides which virtual machine is to be instantiated. It also decides which nodes are to be operated.

The problem of random load in the cloud computing is further improved by the implementation of Equally Spread Active Execution (ESAE) algorithm [34]. In the proposed algorithm as the tasks are submitted, they are queued. If the task size and the size of the virtual machine match, the job is assigned by the job scheduler based on the priority. This algorithm thus improves the response time. Due to equal spread of the job, overall cost is reduced.

Tin-Yu Wu et al. proposed an algorithm called Index Name Server (INS) which aims to minimize the data duplication [35]. In this algorithm an optimal selection point is calculated. The selection depends on several different parameters including position of the server, maximum bandwidth, hash code of the data, weight factor etc. The other main parameter is to check whether the connection can handle additional nodes or not

means the busy level. The busy level if further classified into three parts- First part is the connection is busy and cannot able to handle more incoming nodes. Second part is the connection is not busy and additional nodes can be added. Third is the connection is limited. The limitation of this approach is that it does not predict the future behavior of the nodes.

Rich Lee et al. proposed Weighted Least Connection (WLC) [36] to find out the node with the least number of the connections. If the node with the least number of connections is found, the task is then assigned to it. But the main limitation of this algorithm is that it does not taken into consideration certain factors including the processing speed, bandwidth etc. This limitation is further improved by an algorithm called Exponential Smooth Forecast based on Weighted Least Connection (ESWLC). ESWLC takes into consideration the time series and the node capabilities [37]. This is done with the help of making decision on the basis of CPU power, number of connections, memory, etc. The selection of the node is based on the exponential smoothing.

Al-Jaroodi et al. proposed Dual Direction FTP algorithm [38]. It works by splitting the file of size "n" into "n/2" partitions. Task is assigned to the nodes and processing is done in terms of blocks. The nodes work independently. For example a node starts processing form the first block and continues its processing incrementally while the another node starts the processing from the last node and it does its processing decrementally. Both the nodes will end up processing the whole system in this manner. This reduces the overall response time. This algorithm also reduces the network overhead as it reduces the communication between the client and the nodes. So network load is also taken into consideration in this algorithm. The main limitation of this approach is that full replication of the files is needed; therefore high memory is required for the nodes.

M. Sharma et al. proposed Throttled algorithm [39] to assign the workload on the virtual machines for effective resource utilization. However the response time and resource utilization was improved in 2013 and the modified Throttled algorithm was proposed [40]. The modified algorithm presented the method in which it maintains the index of the virtual machines. The user first request the load balancer to find the appropriate virtual machine. As the request arrives the index first checks the available

virtual machines, and the work is assigned to it. When the next request comes, Virtual machine with the index next to the already assigned index is selected. But this method does not taken into consideration the priority of the work. A method was proposed [41] to consider the priority of the job. It considers the job waiting in the queue for execution. It has also taken into consideration the amount of the time, the task is waiting in the queue must be minimal.

In 2014, an algorithm was proposed [42] to distribute the workload on the least loaded virtual machines with the purpose to improve the resource utilization. This algorithm helps to find out the least loaded virtual machine and allocate the work accordingly. Suppose there are "n" numbers of users and "y" virtual machines are there for processing. The algorithm works by first passing the request to the load balancer. The load balancer maintains the index of virtual machines. At start, all the virtual machines are free, so they follow the round robin algorithm initially but as the next request comes, the algorithm checks the virtual machine table. If the requested virtual machine is available and is not yet assigned, then the requested virtual machine is immediately granted to it. If it is not available at the moment means if it is already assigned to do other work, then the other next least loaded virtual machine is checked in the table and the workload is granted to it.

Yatendra Sahu et al. proposed Dynamic Compare and Balance load balancing algorithm (DCABA) to optimize the cloud [43]. If the load of current selected node is larger than the other selected node, it transfers the part of the load which is extra to the other node. In the proposed technique, by using compare and balance strategy, equilibrium condition is maintained. So the resource utilization is monitored by using process migration. Here there are two processes of optimization. First one is to optimize the system at the machine level and the second one is to check the threshold value of the user application. The algorithm is divided into the two sections. By checking the current load, we decide which part of the section is to be traversed. So first part is to distribute the load using load balancing technique and second is to reduce the servers to reduce the overall cost of the system. This supports the idea of the green computing. Both the sections are further subdivided. Section 1 determines a condition when the load is more than the threshold value. So this is the condition of overloading. This is further resolved by using load balancing technique to balance the

extra load to the other host. Then we find the machine with the minimum probability of the load so the load can be distributed to that machine. Section 2 determines the condition when the load is below the threshold value. This condition is called under loading. This condition increases the overall cost of the system. This is avoided by applying the server consolidation algorithm in which the load is transferred to another host machine to save the cost. Here the main aim is to find the other machine where the load is transferred.

Gulshan Soni et al. proposed Central Load Balancer algorithm in 2014 [44]. This algorithm distributed the workload among virtual machines and is based on hardware configuration and the computing capabilities. The better and reliable resource utilization is achieved through this load balancing algorithm. Figure 2.3 shows the architecture of the Central Load Balancer.

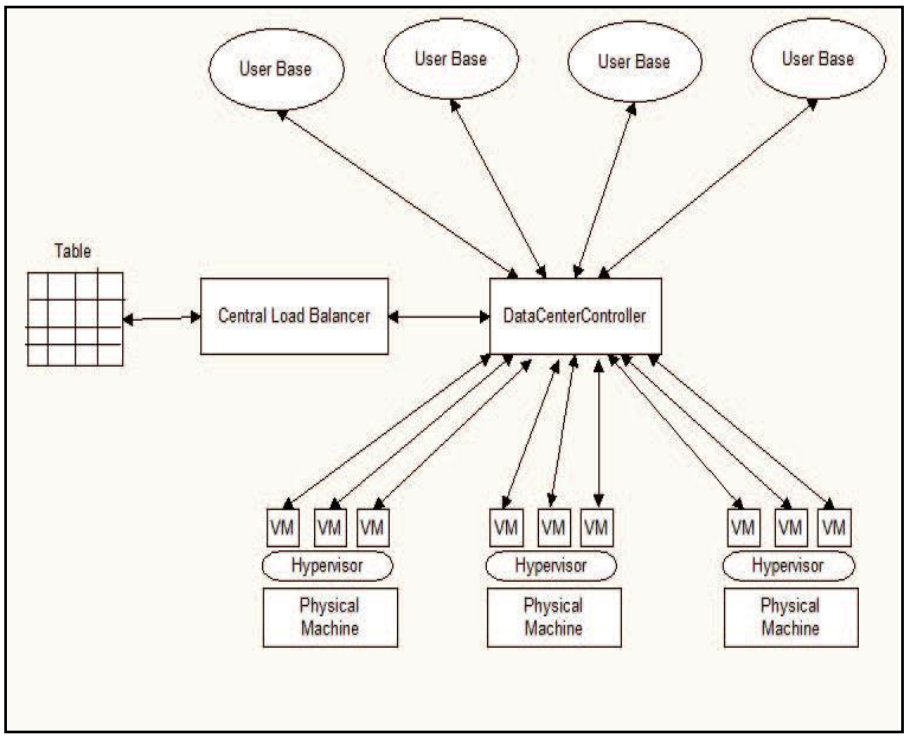

Figure 2.3: Architecture of Central Load Balancer

In this algorithm, first the request arrives at the Data Center. Data Center then asks the central load balancer to allocate the requests. The flowchart of the algorithm has shown in Figure 2.4. It maintains the table that contains the virtual machine ID, states and the set priorities of the virtual machines. States corresponds to the status of the virtual machines either busy or available. The next step is to check the priorities of the

virtual machines so that the work is distributed accordingly. If the virtual machine state is available, then the available virtual machine ID is returned to the Central Load Balancer.

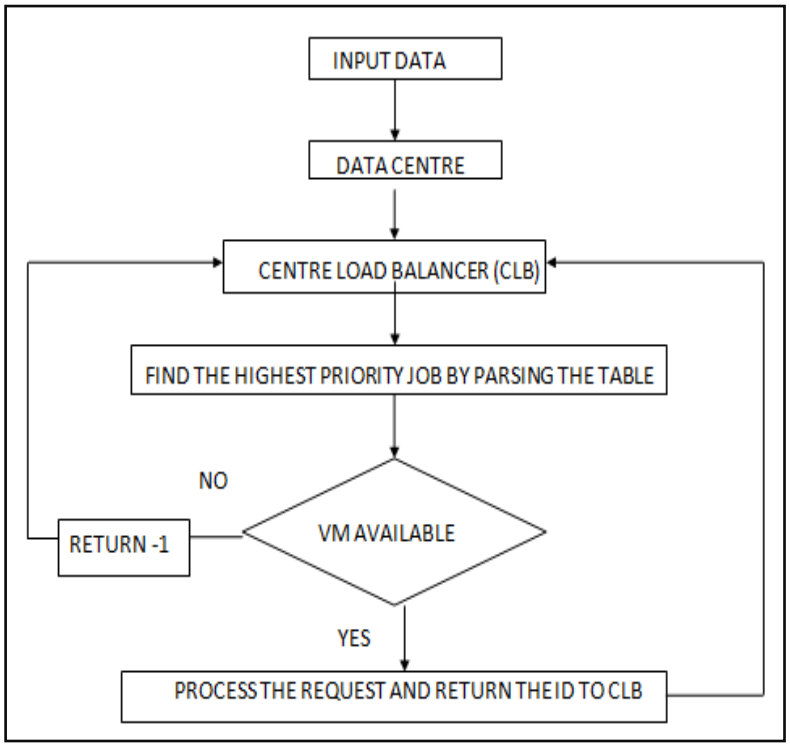

Figure 2.4: Central Load Balancer Flowchart

Load Balancer is responsible for the priority computation. After the allocation of the load to the virtual machine, the table is updated. Data Center gives the notification to Load Balancer controller of the new allocation. If the virtual machine is not available means it is busy, then the ID -1 is returned and the request is queued in the data center controller. Central Load Balancer is linked with the users and the virtual machines. It is used to calculate the priorities and then allocates the work based on the processing speed of the machine. Priority depends on the two factors including processor speed and the memory of the machine. After the virtual machine finishes the request, Data centre gives the notification to the Central Load Balancer of the completion of the request. Now the data centre checks the next request in the queue and is available the above steps is repeated. So 'Central Load Balancing' aims to improve the resource utilization by avoiding the problem of under loading and the over loading. This algorithm efficiently shares the load among the virtual machines.

Shridhar G.Damanal et al. proposed optimal VM Assign Load Balancing Algorithm for efficient utilization of virtual machines to distribute and assign the workload on the least loaded virtual machines [45]. It helps to improve the resource utilization. The entire process has shown in Figure 2.5.

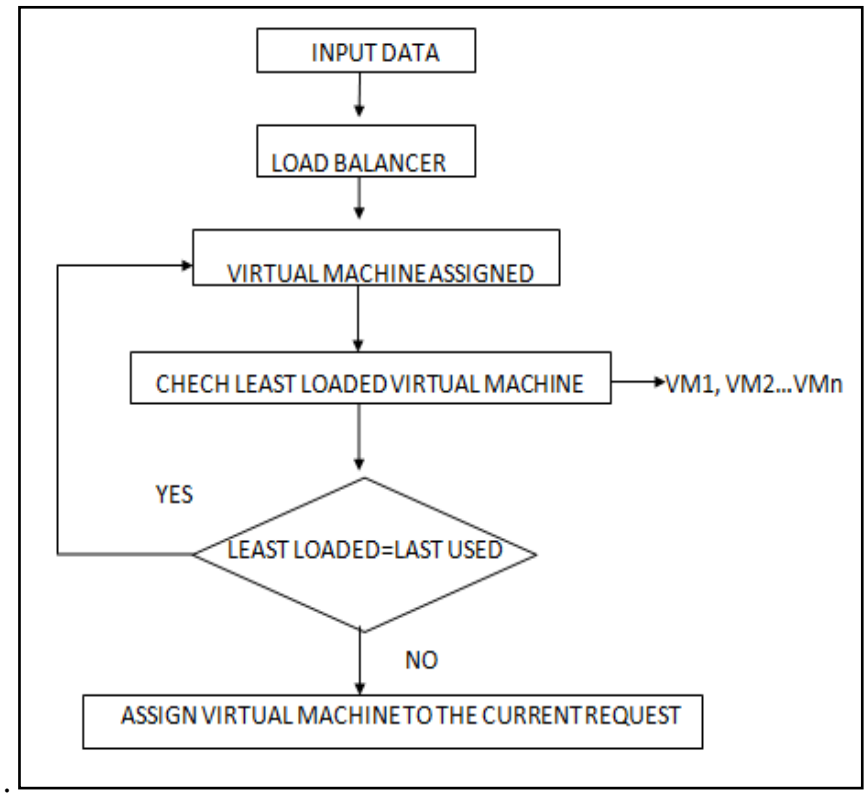

Figure 2.5: VM Assign Load Balancing Flowchart

This algorithm helps to find out the least loaded virtual machine and allocate the work accordingly. The algorithm works by first passing the request to the load balancer. The load balancer maintains the index of virtual machines.

At start, all the virtual machines are free so they follow the round robin algorithm initially but as the next request comes, the algorithm checks the virtual machine table. If the requested virtual machine is available and is not yet assigned, then the requested virtual machine is immediately granted to it. If it is not available at the moment means if it is already assigned to do other work, then the other next least loaded virtual machine is checked in the table and the workload is assigned.

2.5. Comparison on various Load Balancing Algorithms

Table 4 shows the comparison with various Load Balancing Algorithms. This table covered various advantages and limitations of these algorithms.

Techniques	Advantages	Limitations
Round Robin	It selects the first node randomly and then allocates jobs to all other nodes in a round robin manner	Some nodes might be heavily loaded and some are not. Since the running time of any process is not known prior to execution, there is a possibility that nodes may get heavily loaded.
Weighted Round Robin	In this algorithm, a weight is assigned to each node and depends on the weight, the requests are processed.	As precise prediction of execution time is not possible, therefore this algorithm is not preferred.
Mapreduce Based entity Resolution [23]	Distribute the entities of large blocks among multiple reduce tasks.	High Processing Time
Central Load Balancing Decision Model (CLBDM) [24]	Improvement to the round robin. In this model a connection time between the user and the node is calculated and is compared against the threshold value.	Since this method works based on the round robin technique, if the connection goes above the threshold level problem arises.
Ant Colony Optimization [25]	• Under loaded node is found at the beginning of the search • It is Decentralized, so there is no single point of failure. • Ants can collect the information faster.	• Due to large number of ants, network may be congested. • The status of the nodes after the ant's visit is not taken into consideration.
Load Balancing Min-Min (LBMM) [26]	Job with the smallest time is executed first.	The drawback of this algorithm is that some jobs may experience starvation.
Load Balancing Max-Min- Max OLB + LBMM [28]	Uses opportunistic Load Balancing (OLB) to keep each node busy and uses Load Balancing Min-Min (LBMM) to achieve minimum execution of each task.	Response time can be improved
Honeybee Foraging	Achieves global Load	The computation of the

Behavior [29]	Balancing through local server and used for large scale cloud systems	profit may cause an additional overhead which results in overall decrease in the throughput.
Biased Random Sampling [30]	Achieves Load Balancing across all system nodes using random sampling of system domain	The computation of the walk length creates an additional overhead.
Active Clustering [31]	Optimizes the job assignment by connecting similar services.	This degrades its performance when increase in diversity of nodes.
Join-Idle-Queue [32]	First assigns idle processors to dispatchers for the availability of the idle processors at each dispatcher. Then assigns jobs to processors to reduce average length of jobs at each processor.	It cannot be used for today's dynamic-content web services due to the scalability and reliability.
Power Aware Load Balancing (PALB) [33]	PALB algorithm was designed with the purpose to give computation control to the cluster controller	Overloaded,underloaded case isn't considered.
Equally Spread Active Execution (ESAE) algorithm [34]	In the ESAE algorithm, as the tasks are submitted, they are queued. If the task size and the size of the virtual machine match, the job is assigned. This is done by the job scheduler based on the priority.	Power consumption case isn't considered.
Weighted Least Connection (WLC) [36]	It finds out the node with the least number of the connections. If the node with the least number of connections is found, the task is assigned to it.	It does not takes into consideration important factors including the processing speed, bandwidth etc.
Exponential Smooth Forecast based on Weighted Least Connection (ESWLC) [37]	• More Accurate results than WLC • Takes into consideration two main factors, time series and node capabilities.	• Complicated • Prediction algorithm. requires existing data and has long processing time
Dual Direction FTP (DDFTP) [38]	DDFTP is as Fast process.	The main issue is due to full replication of the data files, high storage capacity is needed.
Throttled Algorithm[39]	Assign the workload on the virtual machines for effective resource utilization	Failed to distribute load uniformly, overloading initial VMs and leaving

		others underutilized
Modified Throttled Algorithm [40]	Improved resource Utilization as compared with the Throttled algorithm.	This method does not taken into consideration the priority of the work.
Optimal Virtual Machine Assign Load Balancing [42]	• This algorithm helps to find out the least loaded virtual machine and allocate the work accordingly. • The load balancer maintains the index of virtual machines.	The proposed algorithm can still be improved by taking some more dynamic situations of the incoming requests and how the algorithm responses if we mix both static and dynamic loads.
Dynamic Compare and Balance Algorithm (DCABA) [43]	In the proposed technique, by using compare and balance strategy, equilibrium condition is maintained.	Cloud application service running on any host in a datacenter may change their resource requirement.
Central Load Balancer [44]	This algorithm efficiently distributes the workload among virtual machines, based on hardware configuration and the computing capabilities. The load is balanced based on the basis of priority calculated considering hardware parameters.	It has considered memory and the speed as the parameters for priority calculation. Other factors may also take into consideration to improve the utilization.

Table 4: Comparison on various Load Balancing Algorithms

Chapter 3

PROBLEM STATEMENT

Problem Definition- In cloud computing, there is an issue regarding overloading of the tasks due to the random arrival of the tasks. Due to the random utilization of the CPU, resources are sometimes heavily loaded whereas the other resources are idle. Load Balancing is a way to distribute the entire load over the network across a large number of virtual machines or CPU. This helps to achieve the balanced utilization which maximizes the performance and minimizes the response time. So there is an important issue to balance the load efficiently among the resources.

3.1. Research gap

In Throttled Load Balancing algorithm, only pre defined number of the tasks is allocated. The problem occurs if the request arrives more than the pre defined number of the tasks. Round Robin also handles the problem in the similar fashion. In that situation, the requests have to be queued till the VM becomes available. In active monitoring load balancing algorithm, the work is allocated by distributing the entire workload equally to the available virtual machines. The main limitation of these approaches is that if the hardware configurations of these machines are different then how to balance the load. The Central Load Balancer distributes the work to the virtual machines according to the priority and the states of the machines. But it does not take into account the current utilization of the nodes. However the problem of under loading and overloading is solved to some extent and also the response time of the algorithm is less as compared with the other algorithms.

The main challenge is to develop an approach that will be dynamic and avoids the situation of over loading and under loading. The proposed work is executed on the simulator known as CloudSim and its toolkit. Our research not only implements load balancing algorithm based on resource utilization but also analysis various Load Balancing Algorithms which helps to balance the workload in an efficient manner. Load Balancing based on Resource Utilization shares the load effectively by considering priority policy based on memory resource, CPU speed and the power consumption which is being an important parameter.

Chapter 4

PROBLEM FORMULATION

An efficient load balancing algorithm based on resource utilization is proposed for efficient resource utilization and it also avoids overloading and underloading condition. This load balancer is connected to the datacenter and all its users. All the virtual machines are managed by the datacenter controller. Load Balancer computes the priorities of the virtual machines by parsing its table, based on their speed, memory and power consumption. Load balancer then passes the load to highest priority virtual machine. Due to efficient utilization of the resources, this algorithm is also energy efficient.

4.1. An Efficient Load Balancing based on Resource Utilization

The proposed Load Balancing Algorithm based on Resource Utilization description is as follows-

1. Initially all the virtual machines have no allocations.
2. Datacenter Controller receives the user request.
3. The request is passed to the load balancer.
4. Load Balancer maintains the table which contains Virtual machine ID, speed of the virtual machine, memory resource and the power consumption of the virtual machine.
5. Load balancer parses the entire table which contains the hardware configurations include speed, memory and the power consumption.
6. After parsing the table from top to bottom, it calculates the priority of all the virtual machines from top to bottom.

   Priority(i) = s(i) + m(i) + (1/P(i)), where i=1<n and Priority lies between [0,1]

   where s= speed of CPU

   m = memory resource

   P = Power consumption

   n= number of virtual machines
7. If utilization of vm > threshold, then the process will enter into queue, goto step 2.

8. The highest priority virtual machine is selected and its ID is returned to the Datacenter controller.
9. Datacenter processes the request and load balancer updates the table accordingly.
10. The Data Center Controller checks if there are any waiting requests in the queue. If there are, it continues from step 3.

4.2. Activity Diagram of Load Balancer

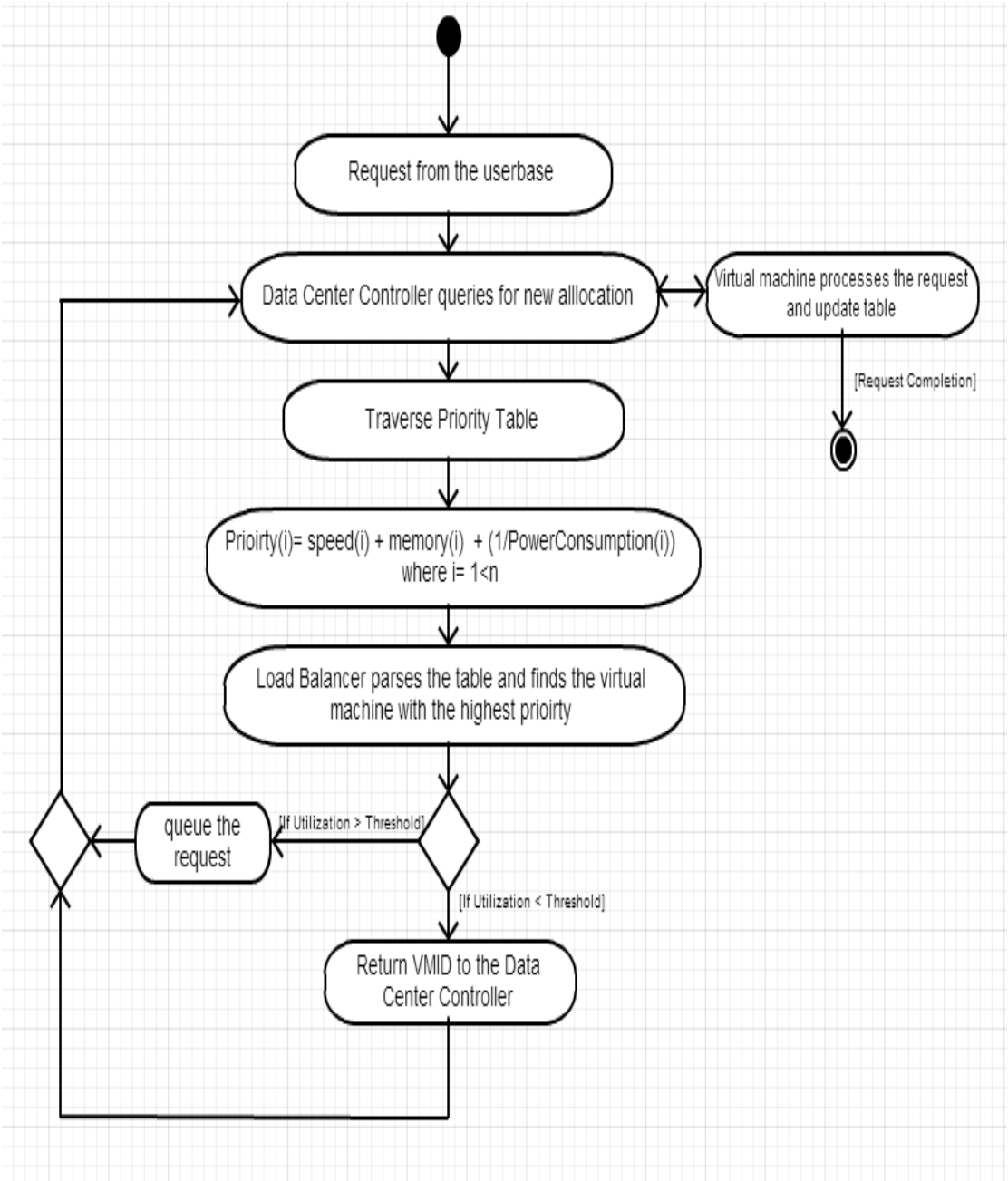

Figure 4.1: Activity Diagram

4.3. Power Consumption Model

CPU consumes the main energy. Power consumption can be calculated by considering CPU utilization. Idle server consumes 70 percent of the total power, consumed by CPU running at the full speed. The power model states [46] that

$$P(util) = f * P_{max} + (1 - f) * P_{max} * util$$

where

P_{max} =Power Consumption at the peak level (Full Utilized)

util = CPU Utilization

f= the fraction of power consumed by the idle server

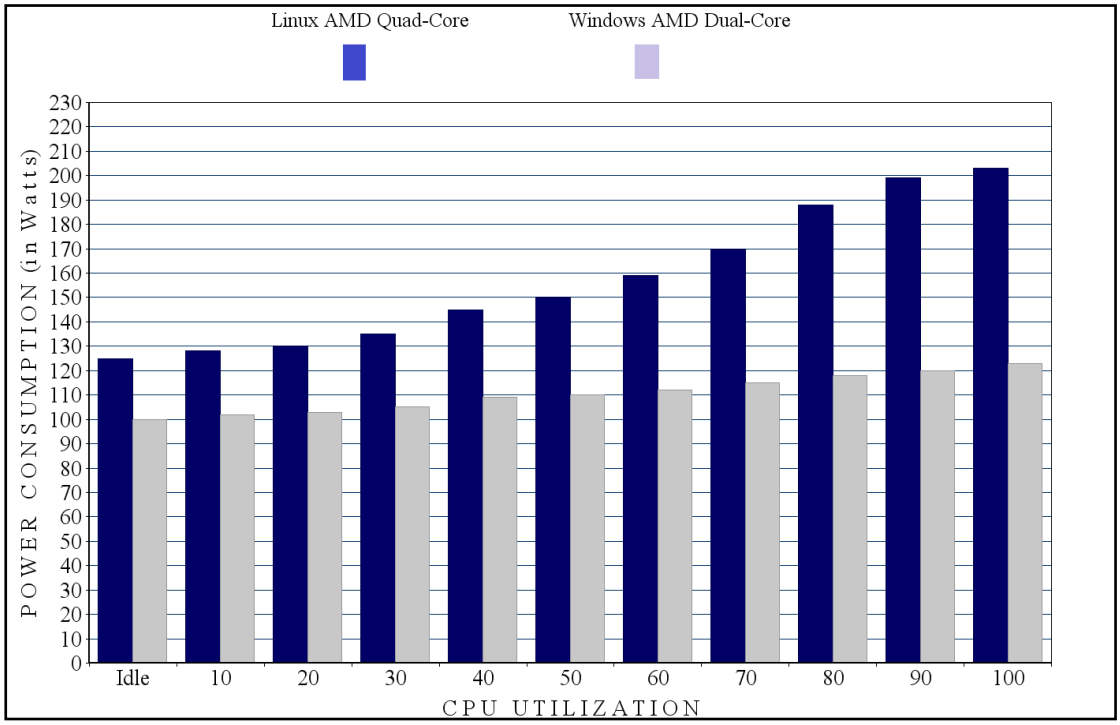

Figure 4.2: Power Consumption Model [46]

For the experimental setup, this thesis has considered, P_{max} is 250 W. According to Standard Performance Evaluation Corporation (SPEC) power, 259W was the average power consumption by servers at 100% utilization. CPU utilization changes with the change in time. Therefore the CPU utilization is a function of time and the energy consumption is calculated as integral of power consumption.

4.4. Cloud Simulation

Cloudsim is a simulator [47][48] that enables to instantiate more than 10.000 machines in less than 5 seconds and that too with the help of only 75MB of RAM [49]. It mainly extends the functionalities of the basic GridSim. It has been extended by various research projects. The extended functionalities include resource provisioning between the virtual machines, modeling storage and the application servers, simulation of the cloud etc. CloudSim is used in many research fields. The research fields include energy efficient cloud scheduling algorithms [50], and further enhancement in the communication flow includes the green computing energy scheduling algorithms [51]. The CloudSim is further enhanced to support the user's perspective. The CloudSim is also extended for the education purposes and the resultant is the TeachCloud [52] which provides the graphical interface so that the students can perform the experiments on the cloud easily.

CloudSim provides the environment for load balancing algorithms. It basically allows the virtual machines to be managed by the hosts. The management of the hosts is done by the datacenters. A cloudsim basically contains four entities which are able to manage the cloud resources. These four entities include Datacenters, Host, Virtual Machines, Applications. Hosts are defined as the pre configured servers with several processing capabilities. Datacenter provides the infrastructure services to the cloud. It also acts as a home to several hosts and the aggregation of all the hosts lead to datacenter entity. Host provides services to cloud. They have their own memory and the storage. Speed of the processing is represented by Million Instructions Per Second (MIPS). Host act as a home to several virtual machines and the aggregation of all the virtual machines lead to a host entity. Virtual machines are mapped to the hosts. The matching characteristics include requirements such as storage requirement, processing requirement, memory requirement etc. Therefore, the same instances of the virtual machines can be mapped to the same instances of the hosts based on the availability scenarios. The execution of the applications are done on the virtual machines.

CloudSim provides various services as shown in the Figure 4.3. The main elements of the CloudSim are also shown in the figure.

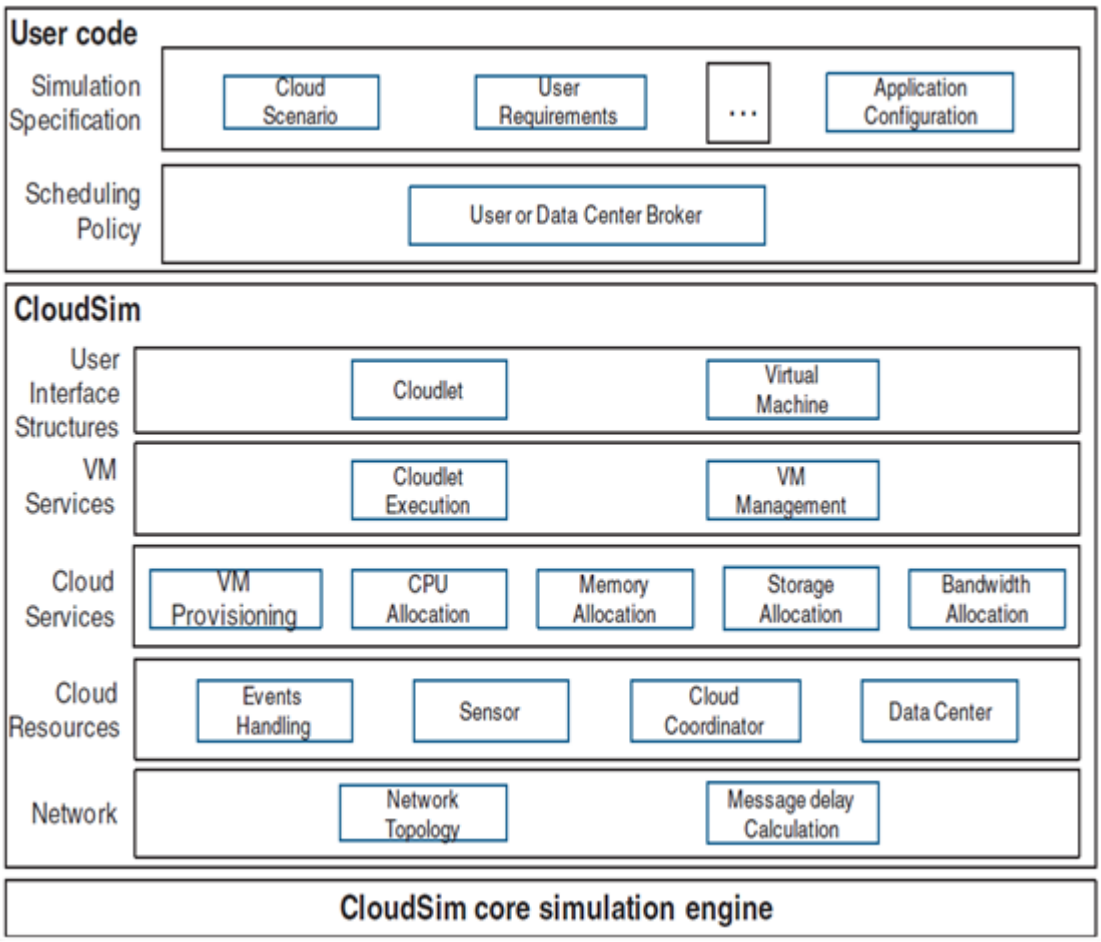

Figure 4.3: CloudSim Layered Architecture [47]

I. Data Center- Data Center creates either the homogenous or heterogeneous data consists of the configuration including memory, storage, capacity etc. It consists of a number of the hosts.

II. Virtual Machine- A virtual machine handles the scheduling and sharing policy. A number of the Virtual machines can run on a host simultaneously.

III. Host- The hosts have the allocation policy to distribute the workload across the resources. The hosts have sufficient memory and the bandwidth to process the requests. Virtual machines handle the data to be processed. A host also acts as an agent for the construction and destruction of the virtual machines.

IV. Cloudlet- Cloudlet delivers the data in the cloud. It is a class that contains various jobs/tasks. It consists of IDs of the data transfer components. It is an application component that delivers the services.

Chapter 5

EXPERIMENTAL RESULT

The proposed algorithm is implemented through the simulation software like cloudsim and the cloudsim based toolkit. For the implementation of the application, java language is used. Assuming that the application is deployed in single data center which has a number of virtual machines. The hardware and the software requirements are stated in the section 5.1.

5.1 Hardware and Software requirements

I. Hardware used

1. AMD Processor
2. RAM size 3.0 GB
3. Processor 1.30 GHz.

II. Software used

1. Operating System- Windows 7
2. Java Software version 1.7
3. NetBeans IDE 8.0
4. CloudSim 3.0
5. Cloud analyst toolkit

Load balancing based on resource utilization dynamically allocates the cloudlets to a particular VM which is determined by VM's speed, memory and power consumption. After finishing the processing of the request, it evaluates the start and finish time of the job. The proposed algorithm solves the problem of load imbalance and results in efficient resource utilization.

5.2. Procedure

Step1: Input the task/workload. Submit the job to the load balancer.

Step2: Activate load balancer.

1. Create datacenter, Host, VMs and broker.
2. VMs and cloudlets are stored and submitted to broker.
3. Start the simulation.

Step3: Load balancing.

1. Check Priority table.
2. Priority calculation depends on the hardware parameters including speed, memory and the power consumption. Power consumption depends on the utilization of the resources, which is calculated by using Power Model.
3. Allocates job to the highest priority virtual machine.
4. Repeat job allocation until all the jobs are processed.
5. Stop the simulation.

5.3. Response Time Calculation

The response time, RT is calculated with the help of the following formula-

$RT = Fin_{tm} - Arr_{tm} + TDelay$

where, Arr_{tm} is the arrival time of user request and Fin_{tm} is the finish time of user request.

The transmission delay, TDelay is calculated with the help of using the following formulas

TDelay = Tlatency + Ttransfer

where, TDelay is the transmission delay T latency is the network latency.

Ttransfer is the time taken to transfer the size of data of a single request from source location to destination. TDelay is considered to be same in every case and hence it is considered as zero.

5.4. Load Balancing Execution

The experiment is conducted in cloudsim simulator using java netbeans. The algorithm can also be integrated into cloudanalyst, which is an extension to cloudsim toolkit. Cloudanalyst also provides GUI facility. Virtual machines with different hardware parameters are used in cloudsim.

After the implementation of the algorithm, the experiment steps conducted are as follows-

I. Initial Step Before Allocation of Workload

Data Center Controller is connected to all virtual machines. The data center controller maintains the table that contains virtual machine ID, speed, memory and the power consumption of each virtual machine. Initially the table is empty, before allocation of the workload and all the virtual machines have null allocations. The load balancer is connected to all the virtual machines through the data center controller as shown in the Figure 5.1.

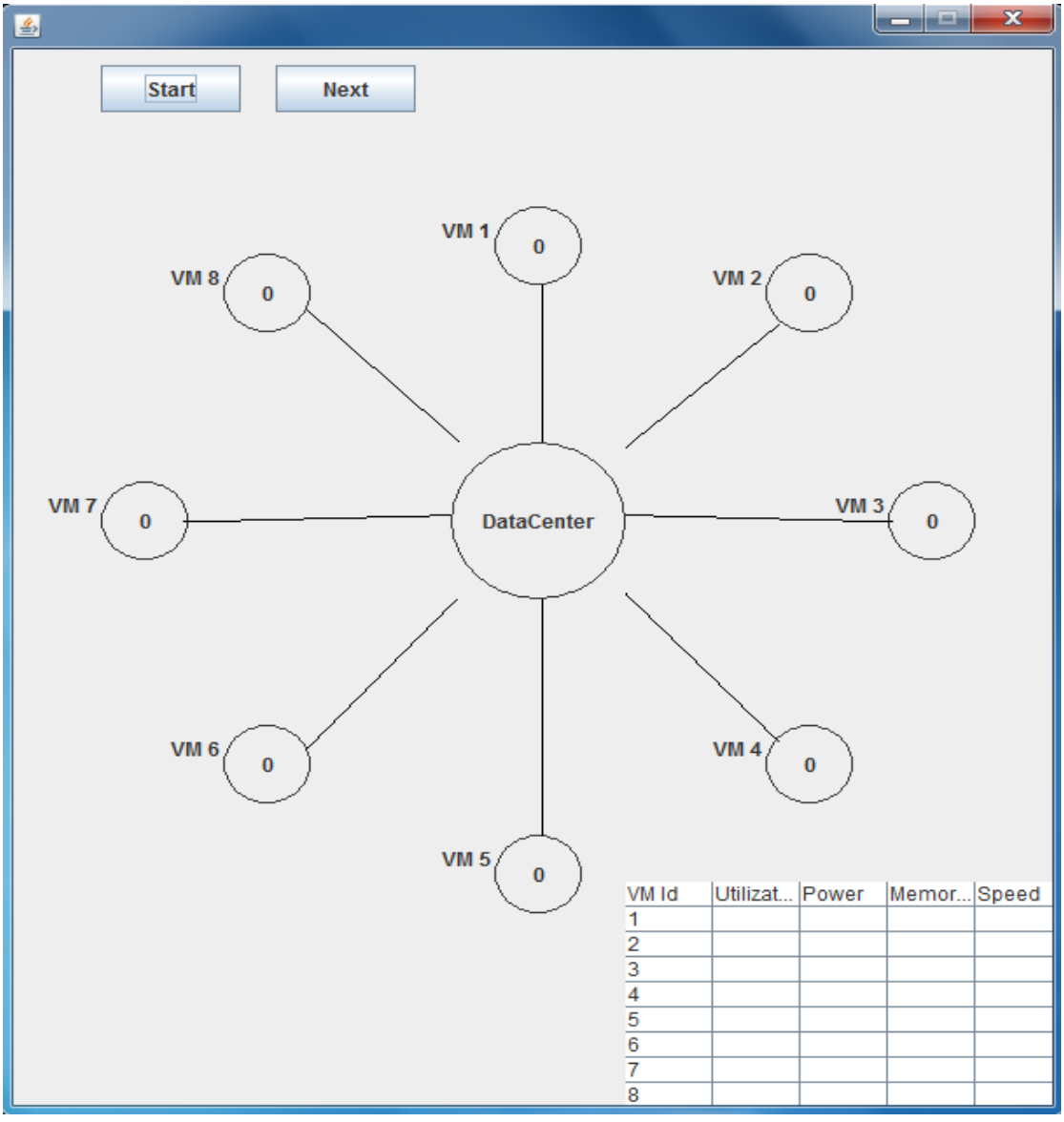

Figure 5.1: Initial Virtual Machines Before Allocation

II. Allocate Task According to Priority of Virtual Machines

Task is allocated to the virtual machines according to the priority, which depends on hardware configuration and the resource utilization. This means, it also checks the least loaded virtual machine for allocation of the task to the virtual machine.

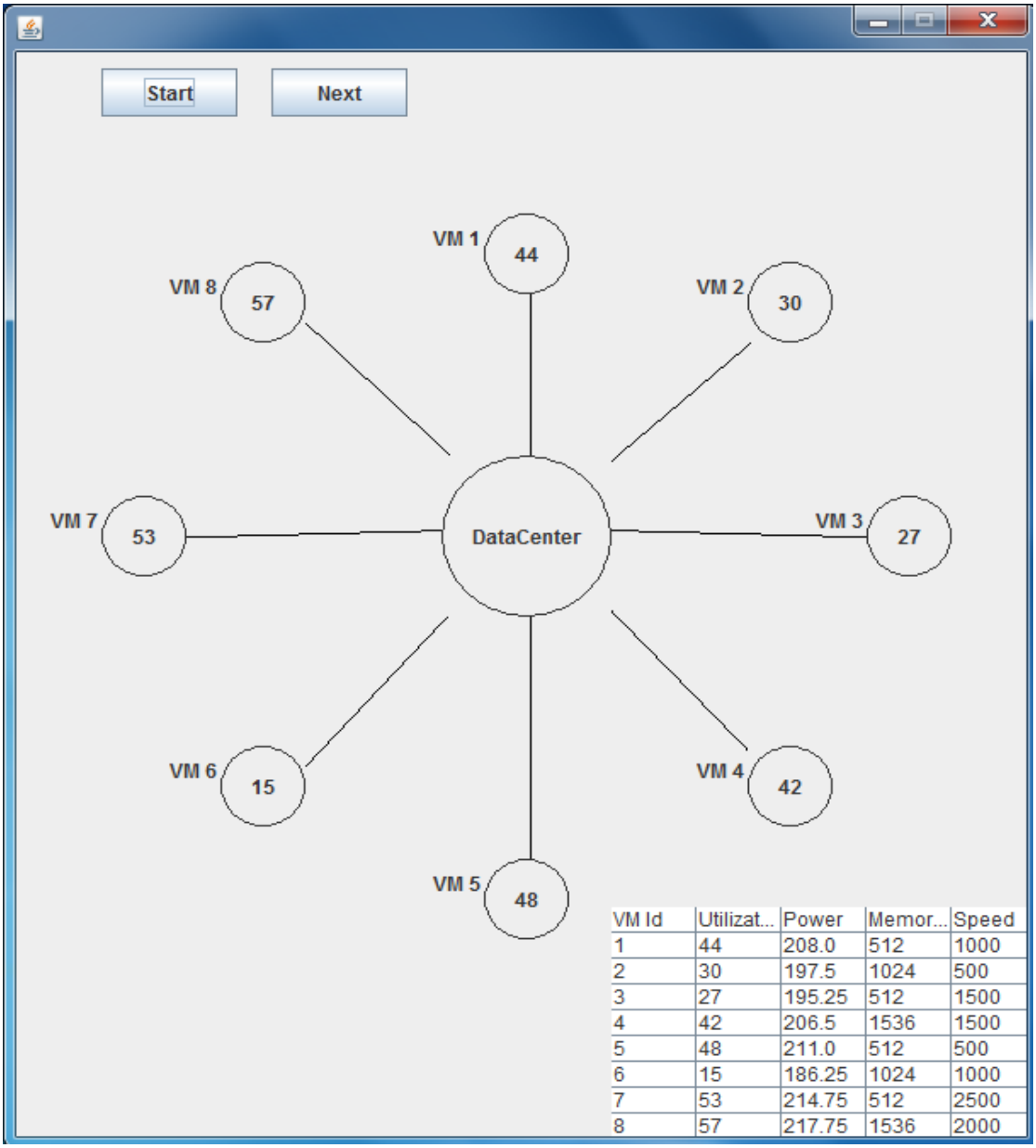

Figure 5.2: Current Status of Load Balancer

III. Details of virtual machine

After the task arrives, load balancer table checks the details of the virtual machine. Figure 5.2 shows the current status of the virtual machines. In, Figure 5.3, table shows current memory, utilization, power consumption and the speed of the virtual machine.

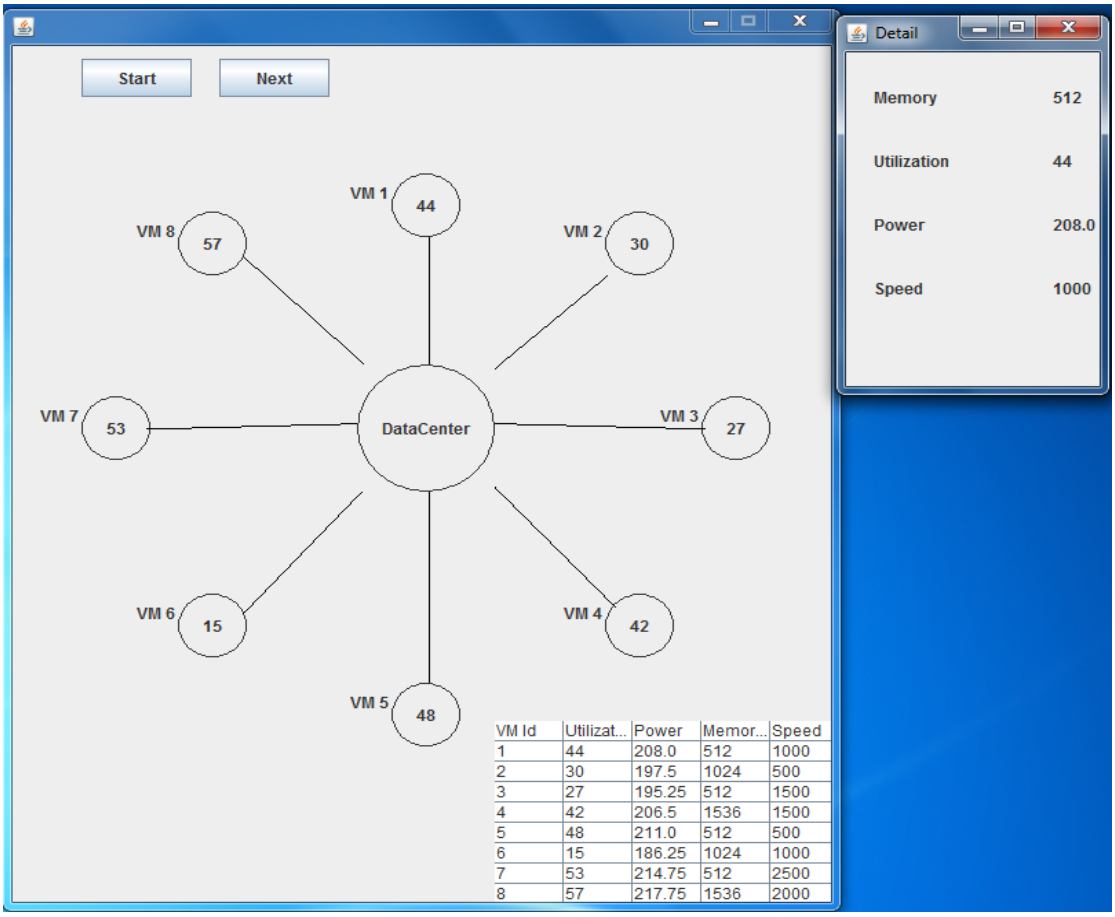

Figure 5.3: Details of a Virtual Machine

IV. Priority Calculation

Priority table contains the priority calculated by considering the speed, memory and the power consumption of the virtual machine. The highest priority is highlighted as shown in the Figure 5.4. It considers 0 as lowest, and 1 as the highest priority.

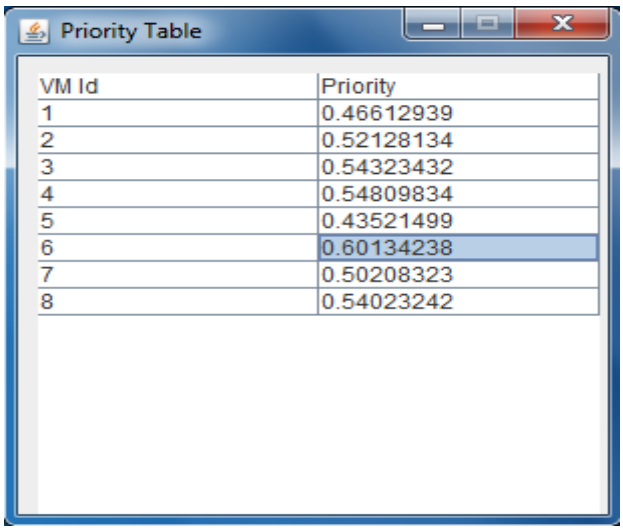

Figure 5.4: Priority table

V. Next highest priority virtual machine is selected

After parsing the table, the work gets allocated to the virtual machine having highest priority as shown in the Figure 5.5. Next highest priority machine is highlighted in the Figure. This VMID accociated with the highest priority virtual machine is then send to the Data Center Controller. After processing the request, the table is updated and Data Center queries for new allocation and the process continues.

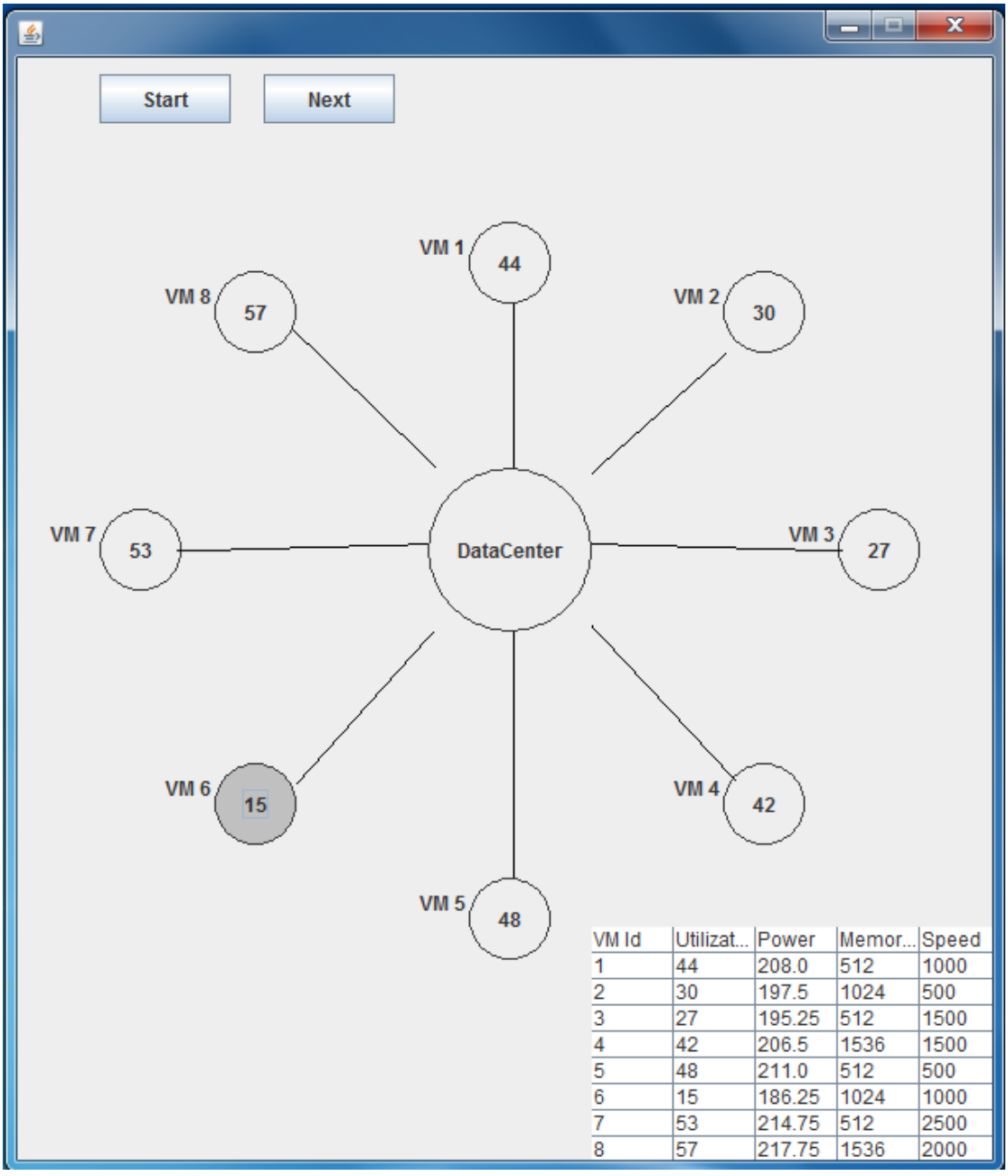

Figure 5.5: Highest Priority virtual machine

5.5. Results

Following is the cloudsim based toolkit parameters configured for the execution of the algorithm as shown in Table 5.

Number of Datacenter	1
Architecture	X86
Operating System	Linux
VMM	Xen
Available Bandwidth per VM	1000
Number of Virtual machines in a datacenter	8
Data size per request (in bytes)	10000
Number of requests	60
Image Size	10000
Physical Hardware Units	4
Instruction length per request(in bytes)	10000
Memory per Host(in Mb)	2048
Storage per Host(in Mb)	100000
Bandwidth per Host(in Mbits/sec)	100000

Table 5: Parameters used for execution

To test the efficiency of the proposed algorithm, Load Balancer based on Resource Utilization, is compared with the other algorithms. The algorithm is compared with the other algorithms using cloud analyst tool. Cloud analyst tool is an extension to the cloudsim. Screenshot of the existing algorithms in the tool has shown in the Figure 5.6.

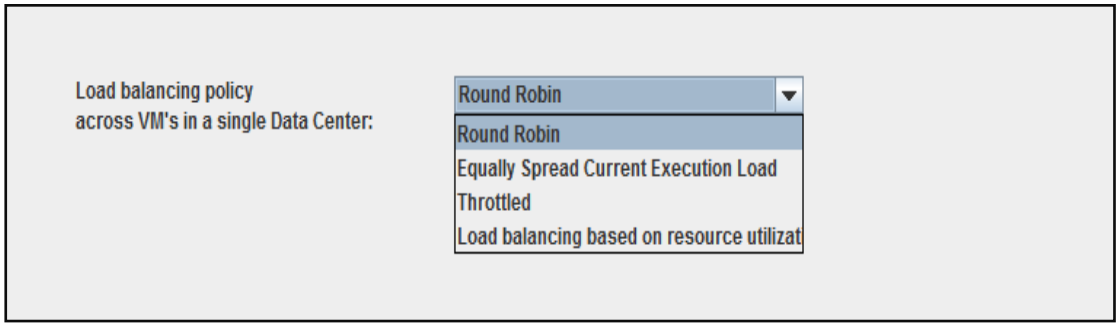

Figure 5.6: Comparison between Load Balancing Policies

Round robin processes the request by selecting the virtual machines randomly and then the workload gets allocated in the circular manner. However it doesn't consider the processing time. A request has to wait in queue if no virtual machine is available. So, this increases the overall response time. Response time of this algorithm with the assigned configuration has shown in the Figure.5.7.

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	311.66	242.92	380.43

Figure 5.7: Round Robin Load Balancer

Equally Spread current Execution distributes the workload equally to all the virtual machines. So in this way equal numbers of the tasks are assigned to each virtual machine. No virtual machine is underutilized. Figure 5.8 shows the response time execution of this algorithm with the configuration as shown in the Table 5.

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	311.12	242.92	380.43

Figure 5.8: Equally Spread Current Execution Load Balancer

Throttled algorithm assigns the workload uniformly to the virtual machines. It takes care of the predefined amount of the tasks are allocated to the virtual machine. Figure 5.9 shows the response time execution of this algorithm with the configuration as shown in the Table 5.

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	310.98	242.92	380.43

Figure 5.9: Throttled Load Balancer

An Efficient Load Balancing based on Resource Utilization algorithm is implemented in cloudsim and its toolkit. This algorithm takes care of the different hardware parameters of the virtual machines. So a priority value is calculated based on the hardware parameters. The result shows that response time has improved as shown in Figure 5.10.

Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	310.12	242.92	380.43

Figure 5.10: Load Balancing based on Resource Utilization

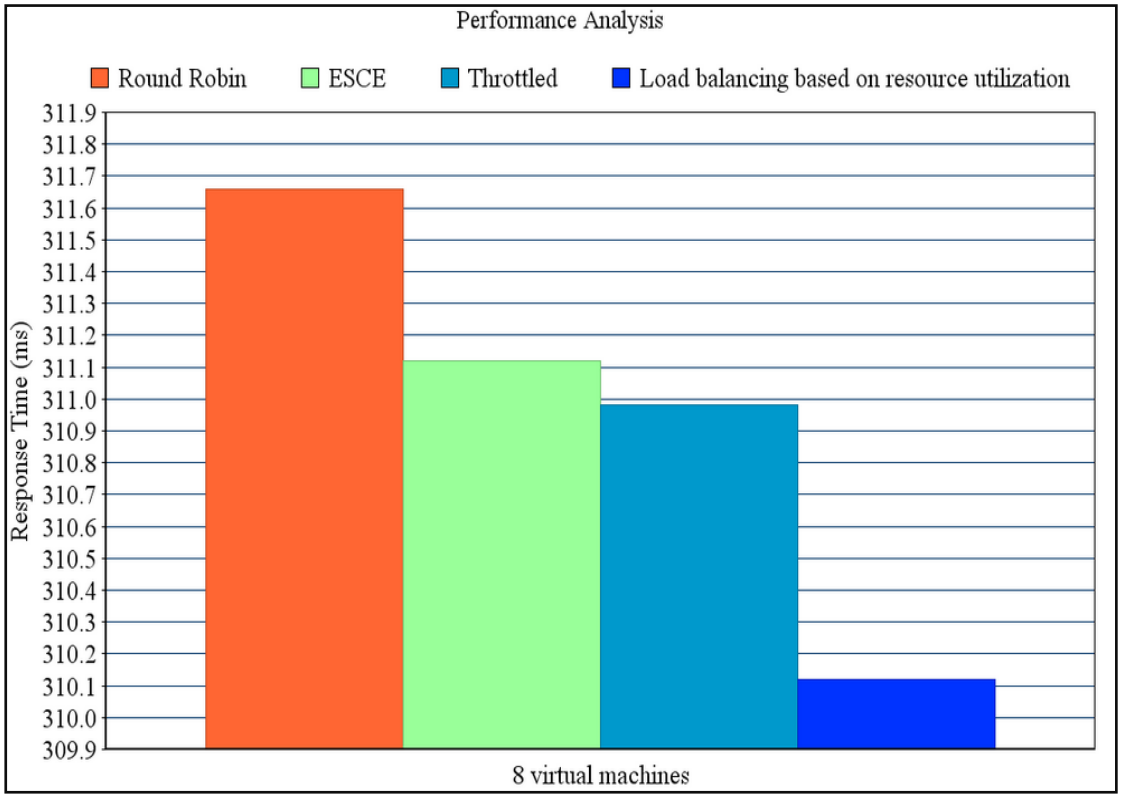

Figure 5.11: Response Time Analysis

The comparison between round robin, equally spread current execution, throttled, and load balancer based on resource utilization has shown in the Figure 5.11. From the analysis of the results, it is proved that the proposed algorithm has better response time as compared to the other algorithms. So this algorithm is response time efficient. Different weights can be assigned to the parameters as follows-

Priority(i) = (a*s(i))+(b*m(i))+(c*(1/P(i)), i=1<n

where s= Speed of CPU

m = Memory resource

P = Power consumed

a= weight of speed

b= memory weight

c= Power weight

n= number of virtual machines

Also a +b +c =1

To calculate the priority of the virtual machines, static weights are applied, however in future dynamic weights can be considered to make load balancing more efficient.

Chapter 6

CONCLUSION AND FUTURE SCOPE

6.1. Conclusion

This thesis explored the recent trends in the cloud computing technology. The main issue remained is load balancing and the resource utilization. Therefore, the load balancing should be more dynamic and efficient to improve the performance of the cloud computing technology. In the load balancing mechanism, as described in the thesis, we have to tackle with the situation of efficient loading of the workload. The existing work considered several load balancing techniques that manage the load distribution among various virtual machines and assigns load corresponding to their priority and states. There is an issue of overloading which means the resources may be over utilized and hence there increases the response time. There is also an issue of under loading means the resources may underutilize and hence there may increase in the power consumption. According to the Load Balancing Algorithm based on Resource Utilization, the workload is distributed to the virtual machine based on different parameters including speed, memory and the power consumption of the virtual machines. The analysis of the results shows that response time of the algorithm is reduced as compared to the other algorithms. A resource allocation policy that takes into consideration resource utilization would lead to a better energy efficiency, as an idle server consume 70% of power with 0% utilization, as per by power model. Hence the proposed work is also energy efficient.

6.2. Future Scope

In future, the load balancing can be more dynamic by assigning the weights to the parameters dynamically in order to to calculate the priority of the virtual machine. In this thesis, response time of the algorithm is improved but the effect of assigning different weight on response time can also be evaluated and the results can be compared in future. So in this way, load balancing can be more dynamic.

REFERENCES

[1] P. Mell and T. Grance, "The NIST Definition of Cloud Computing", *National Institute of Standards and Technology*, Sept. 2011.

[2] E. J. Qaisar, "Introduction to Cloud Computing for Developers: Key concepts, the players and their offerings", in proc. of IEEE *Information Technology Professional Conference (TCF Pro IT)*, pp. 1-6, 2012.

[3] M. B. Mollah, K. R. Islam and S. S. Islam, "Next Generation of Computing through Cloud Computing Technology", in proc. of 25th IEEE *Canadian Conference on Electrical and Computer Engineering (CCECE)*, pp.1-6, 2012.

[4] J. Harauz, L. M. Kaufinan and B. Potter, "Data Security in the World of Cloud Computing", in proc. of IEEE *Security & Privacy*, co published by *IEEE Computer and Reliability Societies*, pp. 61-64, July 2009.

[5] M. D. Dikaiakos, G. Pall, et al. "Cloud computing: Distributed Internet Computing for IT and Scientific Research", in proc. of IEEE *Internet Computing*, pp. 10-13, 2009.

[6] Y. Jadeja and K. Modi, "Cloud Computing - Concepts, Architecture and Challenges", in IEEE proc. of *International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*, pp. 877-880, 2012.

[7] I. M. Khalil, A. Khreishah, S. Bouktif and A. Ahmad, "Security Concerns in Cloud Computing", in IEEE proc. of 10th *International Conference on Information Technology: New Generations (ITNG)*, pp. 411-416, 2013.

[8] [Online: May, 2014] Retrieved from:http://outsideinmarketing.files.wordpress.com/2012/02/cloud_difference_aas.jpg

[9] S. M. Hashemi and A. K. Bardsiri, "Cloud Computing vs. Grid Computing", in *ARPN Journal of Systems and Software*, vol. 2, no. 5, pp. 188-194, 2012.

[10] S. Liu, X. Huang, H. Fu and G. Yang, "Understanding Data Characteristics and Access Patterns in a Cloud Storage System", in 13th IEEE/ACM proc. of *International Symposium on Cluster, Cloud and Grid Computing(CCCG)*, pp. 327-334, 2013.

[11] B. M. Purcell, "Big data using cloud computing", in *Journal of Technology Research*, pp. 1-8.

[12] B. B. Gowrigolla, S. Sivaji and M. R. Masillamani, "Design and auditing of Cloud computing security," in IEEE proc. of 5th *International Conference on Information and Automation for Sustainability (ICIAFs)*, pp. 292-297, 2010.

[13] M. G. Avram, "Advantages and challenges of adopting cloud computing from an enterprise perspective", in Elsevier proc. of 7th *International Conference Interdisciplinary in Engineering (INTER-ENG)*, pp. 529-534, 2013.

[14] A. T. Velte, T. J. Velte and R. Elsenpeter, "Cloud Computing: A Practical Approach", in *Tata Mcgraw-Hill*, 2010 edition, pp. 8-11.

[15] T. Mather, S. Kumaraswamy and S. Latif, "Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance", *O'Reilly Media*, 2009.

[16] E. A. Rayis and H. Kurdi, "Performance Analysis of Load Balancing Architectures in Cloud Computing", in IEEE proc. of *European Modeling Symposium(EMS)*, pp. 520-524, 2013.

[17] R. Buyya, A. Beloglazov, and J. Abawajy, "Energy-Efficient Management of Data Center Resources for Cloud Computing: A Vision, Architectural Elements, and Open Challenges", pp. 1-8, 2010.

[18] K. Nuaimi, N. Mohamed, M. Nuaimi and J. Jaroodi, "A Survey of Load Balancing in Cloud Computing: Challenges and Algorithms", in IEEE *Second Symposium on Network Cloud Computing and Applications(NCCA)*, pp. 137-142, 2012.

[19] A. Beloglazov and R. Buyya, "Energy Efficient Resource Management in Virtualized Cloud Data Centers", in IEEE proc. of 10th *International Conference on Cluster, Cloud and Grid Computing*, pp. 826-831, 2010.

[20] F. Ali and M. Alakeel, "A Guide to Dynamic Load Balancing in Distributed Computer Systems", in *International Journal of Computer Science and Network Security (IJCSNS)*, pp. 153-160, June 2010.

[21] T. Gunarathne, T. Wu, J. Qiu and G. Fox, "MapReduce in the Clouds for Science", in IEEE proc. of 2nd *International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 565-572, Nov. 2010.

[22] G. S. Bedi and A. Singh, "Big Data Analysis with Dataset Scaling in Yet Another Resource Negotiator (YARN)", in *International Journal of Computer Applications (IJCA)*, vol. 92, no. 5, pp. 46-50, 2014.

[23] L. Kolb, A. Thor, and E. Rahm, "Load Balancing for MapReduce based Entity Resolution," in the proc. of IEEE 28th *International Conference on Data Engineering (ICDE)*, pp. 618-629, 2012.

[24] K. Radojevic, B. Zagar and M. Zagar, "Analysis of issues with load balancing algorithms in hosted (cloud) environments", in IEEE proc. of 34th *International Convention on MIPRO*, pp. 416-420, 2011.

[25] N. Kumar, P. Sharma, et al. "Load Balancing of Nodes in Cloud Using Ant Colony Optimization", in IEEE proc. of 14th *International Conference on Modeling and Simulation*, pp. 3-8, 2012.

[26] S. Wang, K. Yan, W. Liao and S. Wang, "Towards a load balancing in a three-level cloud computing network," in the proc. of IEEE 3rd *International Conference on Computer Science and Information Technology (ICCSIT)*, pp. 108-113, July 2010.

[27] J. Ni, Y. Huang, Z. Luan, J. Zhang and D. Qian, "Virtual machine mapping policy based on load balancing in private cloud environment", in IEEE proc. of *International Conference on Cloud and Service Computing (CSC)*, pp. 292-295, Dec. 2011.

[28] C. L. Hung, H. H. Wang, and Y. C. Hu, "Efficient Load balancing Algorithm for cloud computing network", in *International Conference on Information Science and Technology (IST)*, pp. 28-30, April 2012.

[29] M. Randles, D. Lamb and A. Bendiab, "Experiments with Honeybee Foraging Inspired Load Balancing", in IEEE proc. of 2nd *International Conference on Developments in eSystems Engineering(DESE)*, pp. 240-247, 2009.

[30] M. Randles, D. Lamb and A. Bendiab, "A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing", in IEEE proc. of 24th *International Conference on Advanced Information Networking and Applications Workshops*, pp. 551-556, 2010.

[31] O. A. Rahmeh, P. Johnson and A. T. Bendiab, "A Dynamic Biased Random Sampling Scheme or Scalable and Reliable Grid Networks", in *INFOCOMP - Journal of Computer Science*, vol.7, no.4, pp. 1-10. Dec. 2008.

[32] Y. Lua, Q. Xiea, G. Kliotb, et al. "Join-Idle-Queue: A Novel Load Balancing Algorithm for Dynamically Scalable Web Services", in 29th *International*

Symposium on Computer Performance, Modeling, Measurements and Evaluation, pp.1056-1071, 2011.

[33] J. M. Galloway, K. L. Smith and S. S. Vrbsky, "Power Aware Load Balancing for Cloud Computing", in IEEE proc. of the *World Congress on Engineering and Computer Science(WCECS)*, pp. 19-21, October, 2011.

[34] J. kaur, "Comparison of load balancing algorithms in a Cloud", in *International Journal of Engineering Research and Applications (IJERA)*, vol. 2, issue 3, pp. 169-173, 2012.

[35] T. Y. Whu, W. T. Lee, Y. S. Lin, et al. "Dynamic load balancing mechanism based on cloud storage", in proc. of IEEE *Computing, Communications and Applications Conference (ComComAp)*, pp. 102-106, January, 2012.

[36] R. Lee and B. Jeng, "Load-balancing tactics in cloud," in IEEE proc. of *International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, pp. 447-454, October 2011.

[37] X. Ren, R. Lin and H. Zou, "A dynamic load balancing strategy for cloud computing platform based on exponential smoothing forecast", in IEEE proc. of *International Conference on Cloud Computing and Intelligent Systems (CCIS)*, pp. 220-224, Sept. 2011.

[38] A. Jaroodi, J. Mohamed and N. Mohamed, "DDFTP: Dual-Direction FTP," in the proc. of 11th IEEE/ACM *International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, pp. 504-503, May 2011.

[39] M. Sharma and P. Sharma, "Efficient Load Balancing Algorithm in VM Cloud Environment", M.Tech. Dissertation, Information Technology Department, Dharmsinh Desai University, 2012.

[40] S. G. Domanal and G. R. Reddy, "Load Balancing in Cloud Computing Using Modified Throttled Algorithm", in the proc. of IEEE *Cloud Computing in Emerging Markets (CCEM)*, pp. 1-5, October, 2013.

[41] L. D. Babu and P. V. Krishna, "Honey bee behavior inspired load balancing of tasks in cloud computing environments", in proc. of *Applied Soft Computing*, vol. 13, issue 5, pp. 2292-2303, May 2013.

[42] J. Hu, J. Gu, G. Sun and T. Zhao, "A scheduling strategy on load balancing of virtual machine resources in cloud computing environment", in IEEE proc. of

3rd International Symposium on *Parallel Architectures, Algorithms and Programming (PAAP)*, pp. 89-96, 2010.

[43] Y. Sahu, R. K. Pateriya and R. K. Gupta, "Cloud Server Optimization with Load Balancing and Green Computing Techniques Using Dynamic Compare and Balance Algorithm", in IEEE proc. of 5th *International Conference on Computational Intelligence and Communication Networks (CICN),* pp. 527-531, 2013.

[44] G. Soni and M. Kalra, "A Novel Approach for Load Balancing in Cloud Data Center", in IEEE *International Advance Computing Conference (IACC),* pp. 807-812, 2014.

[45] S. G. Damanal and G. R. Reddy, "Optimal Load Balancing in Cloud Computing by Efficient Utilization of Virtual Machines", in IEEE proc. of 6th *International Conference on Communication Systems and Networks (COMSNETS),* Jan. 2014.

[46] J. Adhikari and S. Patil, "Double Threshold Energy Aware Load Balancing In Cloud Computing", in IEEE proc. of 4th *International Conference on Computing, Communications and Networking Technologies (ICCCNT),* July 2013.

[47] R. N. Calheiros, R. Ranjan, R. Buyya, et al. "Cloudsim: a novel framework for modeling and simulation of cloud computing infrastructures and services", pp. 1-9, 2009.

[48] R. N. Calheiros, R. Ranjan, R. Buyya, et al. "Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms", in *Software: Practice and Experience*, pp. 23-50, 2011.

[49] G. Sakellari and G. Loukas, "A survey of mathematical models, simulation approaches and testbeds used for research in cloud computing", in the proc. of Elsevier, *Simulation Modeling Practice and Theory*, pp. 92-103, 2013.

[50] Y. Shi, X. Jiang and K. Ye, "An energy-efficient scheme for cloud resource provisioning based on cloudsim", in the Proceedings of the Annual *International Conference on Cluster Computing (CLUSTER),* Austin, USA, pp. 595-599, 2011.

[51] T. Duy, Y. Sato and Y. Inoguchi, "Performance evaluation of a green scheduling algorithm for energy savings in cloud computing", in IEEE Proceedings of *International Symposium on Parallel and Distributed Processing, Workshops and PhD Forum (IPDPSW)*, pp. 1-8, 2010.

[52] Y. Jararweh, Z. Alshara, M. Jarrah, et al. "Teachcloud: a cloud computing educational toolkit", in the proc. of 1st *International IBM Cloud Academy Conference (ICACON)*, pp. 237-257, 2013.

LIST OF PUBLICATIONS

I. Gurpreet Singh Bedi, Ashima Singh, "Big Data Analysis with Dataset Scaling in Yet Another Resource Negotiator (YARN)", in *International Journal of Computer Applications (IJCA),* vol. 92, no. 5, pp. 46-50, April 2014. Published by Foundation of Computer Science, New York, USA.

II. Gurpreet Singh Bedi, Ashima Singh, "Dynamic Load Balancing in Cloud Computing: A Big Challenge", in IEEE 6th *International conference on Computational Intelligence, Communication Systems and Networks (CICSYN),* Tetova, Macedonia. [Status: Accepted]

III. Gurpreet Singh Bedi, Ashima Singh, "Load Balancing Issues, Techniques and Challenges in Cloud Computing", in Elsevier proc. of *International Conference on Emerging Research in Computing, Information, Communication and Applications (ERCICA),* 2014. [Status: Accepted & Registered]