Chapter I

Introduction

Chapter 1
Introduction

1.1. Introduction

Cloud computing is the ability of using many computing resources which is including internet applications and storage services. The shared pool of the resources is hosted by the cloud supplier. Elasticity is the main attribute of the cloud computing, means its tendency to grow and reduce the computation as per the requirements [1]. Another attribute is scalable, its mean ability to balance the increased demands of the CPU storage, bandwidth etc. According to the National Institute of Standard and Technology [2], 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 need minimal effort. Cloud Computing delivers the secure access to the applications. But high level security is a challenge for the cloud developers

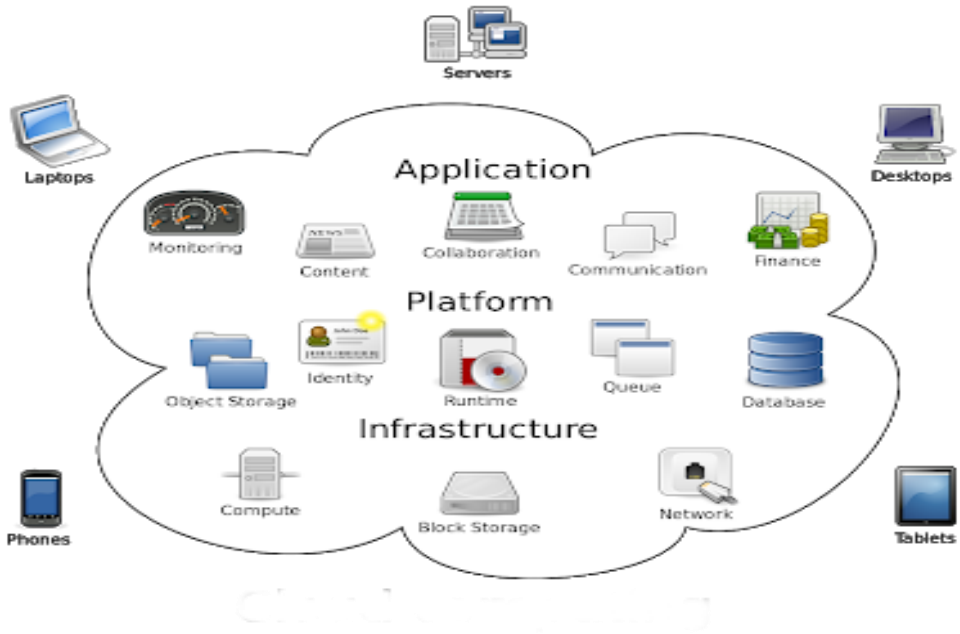


Figure 1.1: Cloud Computing [3]

The traditional computing resources are scalable but are not flexible. 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. John McCarthy in 1960 introduced the concept of the cloud computing and he proposed to have computation- a public efficacy in the future. Prior to the VPN companies are provided the dedicated point to point facility 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 delivers the virtual environment. It is based on Service Oriented Architecture (SOA) to provide hardware support and it transfer the data, processing and service delivery away from the desktop transform to the data centers [4]. It delivers computation, storage and other services that may not need 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 is provide on demand data access to the shared pool of the resources [5]. Cloud computing is related 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. It is distributed into many levels client, server, application, infrastructure and the platform. Modern 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 [6]. This is the main hurdle in the cloud computing. In these issues researchers are working to solve 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. Clam in management- A person needs only internet connection and the web browser to access the facilities 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 main advantage of applications availability.

III. No interruption in the services- Cloud computing delivers uninterrupted services to the user [5].

IV. Disaster Management- In the disaster recovery cloud computing is also efficient. Organizations is need to an important data .An offsite backup is helpful to recover the data. Cloud storage not only keeps the backup of the data offsite and it 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 structure resources. This can be decreased with the help of the cloud computing too few 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 simple to scale up the cloud. The scalability process can be done in less time. Then as well the resources can also scale down with the requirements.

1.2. Cloud Computing Infrastructure

Cloud computing architecture is distributed into two parts- the front end and the back end. The User Interface (UI) is the front end, the client sees and it is responsible for taking the request from the user as an input. The cloud system is the back end; it 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 a management to the system .The networked computer are connected by the middleware.

1.2.1. Layers of Cloud Computing Architecture

The cloud computing are different layers. For first layer is the cloud client consists of the hardware or the software, relies on the cloud to provide the services.

In 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 services (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 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 buy any

server or resources. This also enables fast access to the data with the low cost. Most relevant examples include Go Grid and Flexi scale. Last layer is the server layer which includes the hardware/software helps to deliver 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 managed whereas in SaaS, all the layers are provider managed.

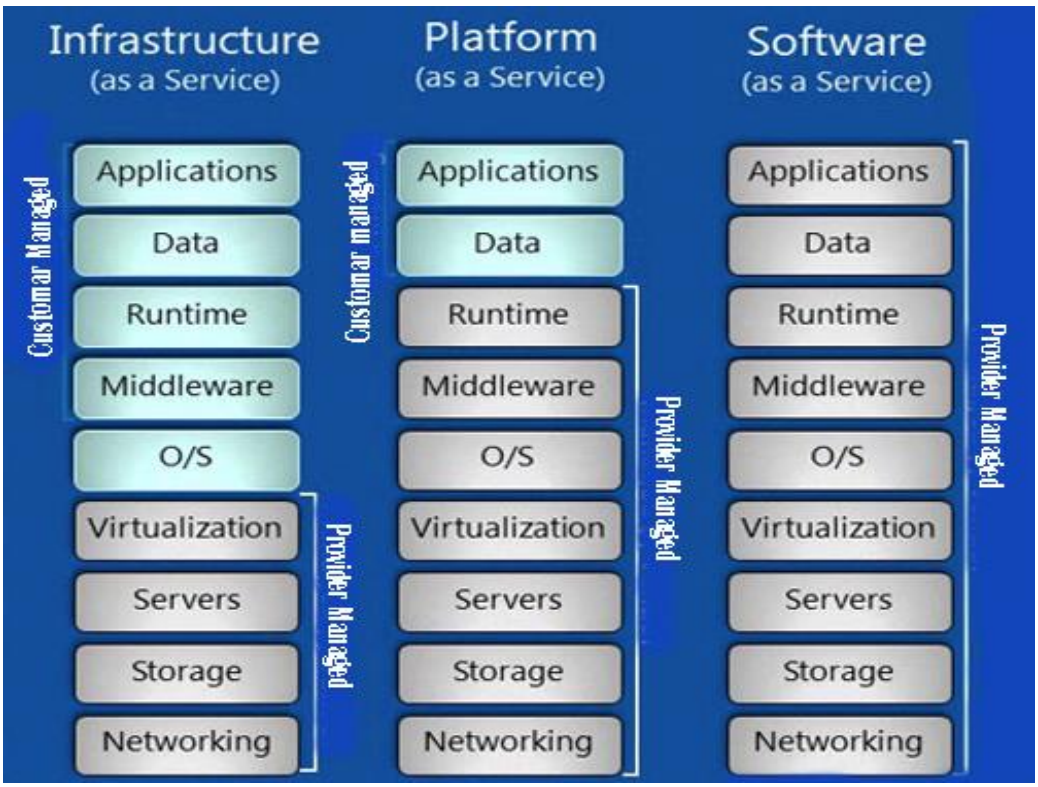


Figure 1.2: Comparison between three main layers of Cloud Computing [7]

Today, the demand of exchanging of the information on the network is continuously increasing. Cloud computing offers IT services to the users globally. This 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 architecture has five layers 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. For example- security concern, resource utilization performance, prone to attack, load balancing in the distributed system etc.

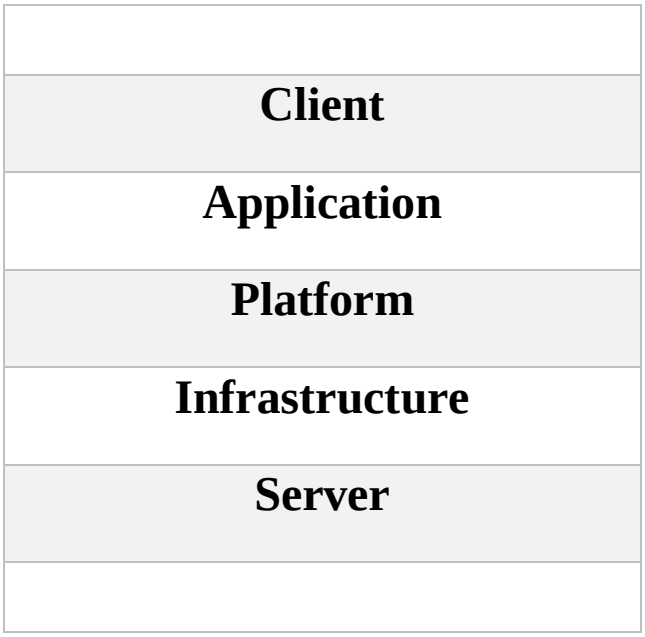


Figure 1.3: Five Layers of Cloud computing

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-Cloud infrastructure can access by an end user. The main of cloud infrastructure to minimize the cost and it is available on-demand basis. The security of the cloud is main challenged. 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 Supplier- cloud supplier 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 supplier 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 supplier and the end user. The developer lies between the two i.e. Cloud End User and the Cloud provider.

Type of stakeholders	Issues
End user	• Ease of use • Security • Privacy • Reduced cast • Availability
Cloud Provider	• Energy Efficiency • Outsourcing • Resource • Meet Requirement • Managing Resources
Cloud Developer	• Virtualization • Availability • Reliability • Data Management • Programmability

Table 2: Cloud Stakeholders

1.2.3 Cloud Deployment

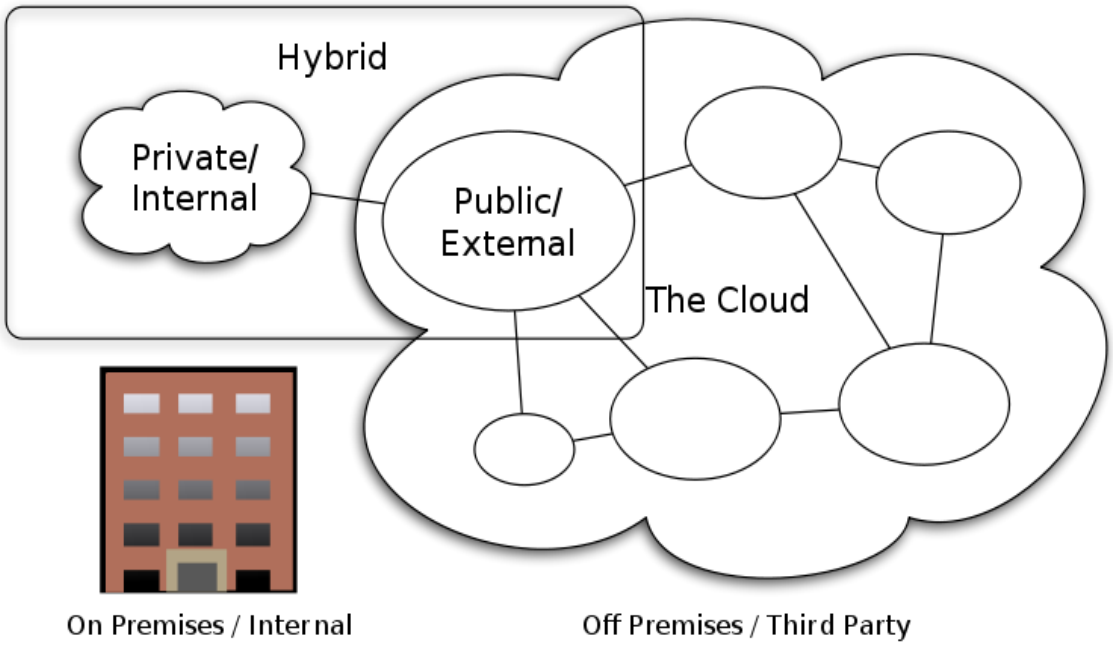


Figure 1.4: Cloud Deployment

i. Empty
ii. Private cloud- All the operations are done by the organization. The main advantage of the private cloud in comparison with the public cloud is that, it is easier to preservation and deployment. Here resources are pooled together and provided at the organization point 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 combination 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.

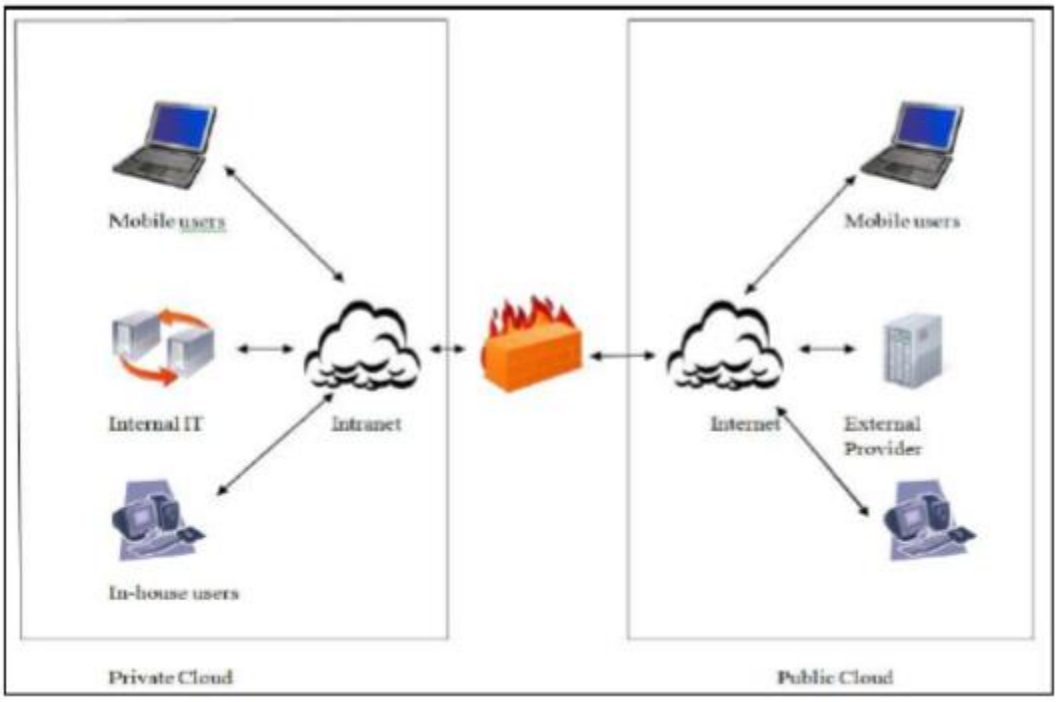


Figure 1.5: An overview of hybrid cloud [8]

iv. Community cloud- The third party or within the organization is provided the community Cloud. A cloud is called community cloud if many organizations share their policies and the necessities in the cloud. So a community cloud is constructed by the many organizations and access to the cloud is done slightly.

1.3. Cloud Computing comparison with Cluster and Grid Computing

The comparable and the dispersed computing is base of the cluster. It combines the resources in the vicinity and thus shares the workload. It takes place under the management of only one organizational area. The combination of the nodes is responsible for the whole computation. In the cluster computing extensive version is grid, as the resources are shared through the internet

geographically. It is the combination of the resources are distributed on the network. The addition of the nodes permits sharing of the resources dynamically. Cloud computing is related 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 field. 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. 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 [8]. So compare to grid computing cloud is more flexible and it provides on-demand resources. In grid computing resource management is always distributed. But in the cloud computing, resource management may either be distributed or centralized.

1.4. Characteristics of Cloud Computing

I. Without the help of any device user can access the data and perform the computations. Cost gets reduced as the infrastructure 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 precise by pay per use facility. It also requires fewer skills for the finishing.
III. Sharing of the resources provides efficient deployment of the bandwidth [9].
IV. The privacy is the main distress in the cloud computing, as the data shared on the cloud may be private. But it provides better security then usual 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 atmosphere on the basis of the three models- public, private and the hybrid. Public cloud means the infrastructure is maintained and owned by the service source. 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 [10].
VII. The services that Cloud Computing helps to delivers were not possible before [11]. For example- Through cloud are location aware and context aware, mobile based applications

provide real time data processing. Through the clouds monitoring applications are also managed. Batch processing enables to process the entire data (size in terabytes) parallel.

Through batch processing Apache Hadoop Mapreduce makes the entire complex process easier. Business forecasters use the large volume of the data to understand the customers. Reduces the latency of bandwidth through the cloud computing also.

1.5. Virtualization

The system of providing the resources is virtualization in cloud computing. For example virtual machines as hardware resources. Virtual machine (VM) is recognized as the guest. The virtual environment responsible for providing by a host. Virtualization gives the delusion of the real thing but it is not real. All the features of the real thing it provides. Just as the real object an end user can use all the services of the virtualized thing. To provide the services to the end user virtualization is related with the cloud. In two ways the datacenter is capable to provide the services either in full virtualization or in Para virtualization [12].Full installation of the machine is done in the full virtualization. The new machine contains all the software of the previous machine after the installation of one machine on the other. Among the multiple users, full virtualization able to delivers the services. It also isolates the users from one other. Among the multiple users a successful sharing is done. Para virtualization is responsible for the care of competent use of the memory and there sources. On the single machine it allows multiple OS to run. In this approach, the services are available moderately. Including disaster recovery is the main advantage of using this approach, capacity management and the migration. Instances of the system are moved to the other machine and the failure is recovered in case of the system failure. Migration of the instances is effortless and easy. In the case of the Para virtualization the power management is also an easy task.

1.5.1. Resource allocation

On the demand basis allocation is to provide the resources. The main aspire is to keep the track of the teeming node so that no wastage of the resources. The wastage implies the wastage of the CPU speed, memory or the bandwidth. In the two levels the entire mappings done - 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 competence and the accessibility. Allocation depends on the on-demand necessities. The second

level is Application mapping to the virtual machine.an application requires some power for execution. 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 accessibility and the arrangement.

1.5.2. Task Scheduling

The scheduling is done after the allocation of the relevant resources. Scheduling specifies the method in which the resources are allocated. Allocation means which resources are obtainable to meet the necessities and the development 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 supply the feature of multiprogramming. In the two ways scheduling is done either as a space shared or as on the time shared. Both the virtual machine and the host are allocated in any way. The resources are preempted in the case of the time sharing mode is the main difference in both the modes. Resources are not preempted in case of the space sharing mode.

There are four cases to be considered:

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

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

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

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

1.6. Load Balancing

A web server has the correspondent to balance the incoming request to the servers. The main intend of correspondent 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 as regards the transfer so that the load is transferred competently to the server which can process the request at that instant. By the front end Web server's load information is used in making the decisions.

To the number of the web servers the front end may end a series of the requests. 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 instant. The server and the front end interactions the information about the load with each other to make the efficient decisions. The service quality is improved and the system becomes more stronger with the help of the correct decision about the load balancing. The load balancer decides how to move on the requests and the decision is made communicate to the CPU load percentage on the particular virtual machine.

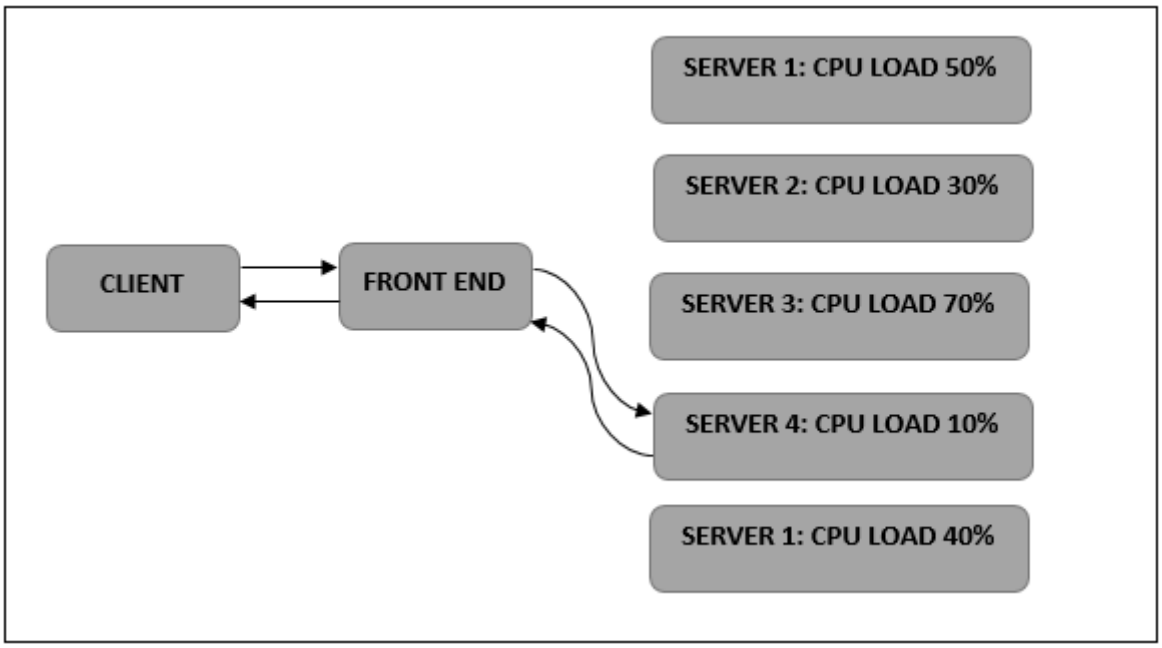


Figure 1.6: Load Balancing

1.7. Motivation

Cloud Computing is a immense research. The internet is viewed as the cloud which provides either connection less or connection oriented services. I have found several issues in cloud computing after studying many research thesis. But the main focus is on load balancing. In cloud computing it is one of the main challenges. To battle with the issues, many load balancing methods has been proposed. The maximize throughput is to the main concern. The main

spotlight in this thesis is to raise the performance of the system by the proper exploitation of the virtual machines. So a new load balancing method is proposed.

1.8. Barriers in Cloud Computing

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

I. Security- At all levels including network security is the main concern, host and the client.
II. Access and Connectivity- Cloud computing depends on the accessibility of the high speed network access. The sharing of the information through the cloud is another aspect that aids in industrialization but it depends on the network access [13].
III. Reliability- Without interruption the applications must be available 24x7 to provide services. In case of the service failure, without any ill effect backup plans must be there and must begin. However to make sure the reliability additional costs may be added to moderate risks.
IV. Interoperability- For the approval of the cloud computing, interoperability between private and public clouds is another concern. It may require additional effort to combine with the traditional applications that may be situated on another cloud depending on the task. The barrier of the cloud computing is maintenance of the interoperability.
V. Economic Cost- Cloud computing is cost effective means of data sharing on the cloud. However there may be hidden cost connected 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 include the risk of increase in cost in change of the solution from one cloud to another.

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

VII. Global Issues and boundaries- There is inconsistency 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 policy may apply in adoption of the technology. This results in negative brunt on cloud computing. For example countries like Canada are making law –USA Patriot Act to ensure the privacy and security of the data [14]. Canada told the organizations not to use the computers globally which are within US borders. Amazon Web services have solved this issue to some amount 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 II

Literature Survey

Chapter 2
Literature Survey

Load Balancing is a system to distribute the workload across different nodes, or other resources like central processing units, network links, etc. The main goal is to achieve efficient resource utilization, increase in the throughput, decrease the response time and minimize the overhead. It also supports to achieve the fairly distribution of the workload. A DNS server is scheduled to translate the address to the specific IP when accessing the specific web services through the addresses. This URL translation is tasked to choice a specific node from the cluster. It is based on the scheduling policy of the web servers. Time is defined to cache all the translations in terms of TTL (Time to Live). The next task is routed to the server when time to cache the translation is expired. Round Robin is the method to implement the load in the rotating method. So, 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 data layer protocol or the network layer protocol. This is finished by the process of routing. Cloud computing delivers a model in which a number of the resources can be shared by on-demand service. The resources effectively share, load balancing plays a vital role, which decides how the request can be processed. The decision effects the sharing of the resources in a many 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 various ways. For example IP tunneling, Network Address Translation (NAT) 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 straight passed to the user. This approach is not popular cause to the security concerns. The load balancing is achieved by the relevant hardware/software, for example multilayer switch supports in load balancing. It is also one main reason of cloud computing as there may occur a situation in which some nodes are busy and at the same time and some nodes are idle. This is increase the overall response time. A mechanism is needed to confirm that the workload across the nodes is evenly distributed. Different type of algorithms are there to serve the efficient load balancing.

2.1. Load Balancing Scenarios Based on the Nature of Nodes

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

I. Distributed balancing

In this method, single node is not responsible for the entire scheduling decision. There is efficient distribution of the tasks across the multiple domains are responsible for the whole function. So, no single node is overloaded. Hence the overhead in the distributed load balancing is less as compared to the centralized load balancing. Some examples of the distributed load balancing are honeybee foraging and biased random sampling.

II. Centralized balancing

In this technique, all the decisions are complete by the single node. This node is responsible for managing the entire network. The procedure may be static or dynamic depends on the requirement specification. This procedure decreases the response time to analyze the different type of resources as the whole management is done by the centralized node. Using this approach has some limitations. 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.

III. Hierarchical balancing

This technique mainly works in the master slave node. A parent node is responsible to balance the nodes. The architecture of hierarchical load balancing is created 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 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. After the execution of the subtasks it takes place. The scheduling of the tasks is created on the workflow based algorithms. These algorithms are use directed and acyclic graph (DAG) as the knowledge base. Workflows are classified into two types including transaction incentive and data incentive. In transaction incentive workflows, multiple instances of the workflow have the similar structure. When the size of the data is large this time use data incentive workflows.

2.1.1. Factors that affect Load Balancing

Response Time- The load balancing algorithms is the architecture which is the main factor that affects the performance. In centralized, decentralized and hierarchical load balancer architecture, with the response time increases, increase in the number of the users. However, compare to the

centralized and decentralized approach hierarchical approach performs better [15]. Therefore, the hierarchical approach takes less time as compared to the other methods. The centralized and decentralized approach shows same response time. Server Load- Server load means the number of the requests can be represented as requests/sec, the system can handle per second. The main goal is to split the workload across the three architectures to check their ability to handle the workload. Both decentralized and the centralized load balancers, the servers show the same load with the increase in the number of the nodes. The experiment was performed on the three architectures [15] and the results showed that the performance of the hierarchical load balancer is far better because of its architecture and the ability of the load balancing algorithm to maintain the centralized management. Load balancing technique in the cloud computing supports to maintain the distribution of the workload across the nodes and thus decreases the amount of energy consumed. It helps in avoiding the overheating of the cluster. The consumer, Virtual machine, Resource allocator and the physical machine are the four main element the energy efficient cloud contains [16]. These four elements are shown in the Figure 2.1 and are described below-

I. Consumer

Company that runs through the internet can act as a consumer. This application can be distributes the workload as per the requirement. Requirement is created on the number of the users using the application.

II. Resource Allocator

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

III. Virtual Machines

On the request virtual machines (VM) can be started or stopped. On the single physical machine various VM can run parallel and thus provides the suppleness to update the data resources. The

nodes can be put together to save the energy resources, by the relocation of the virtual machine across the physical machine.

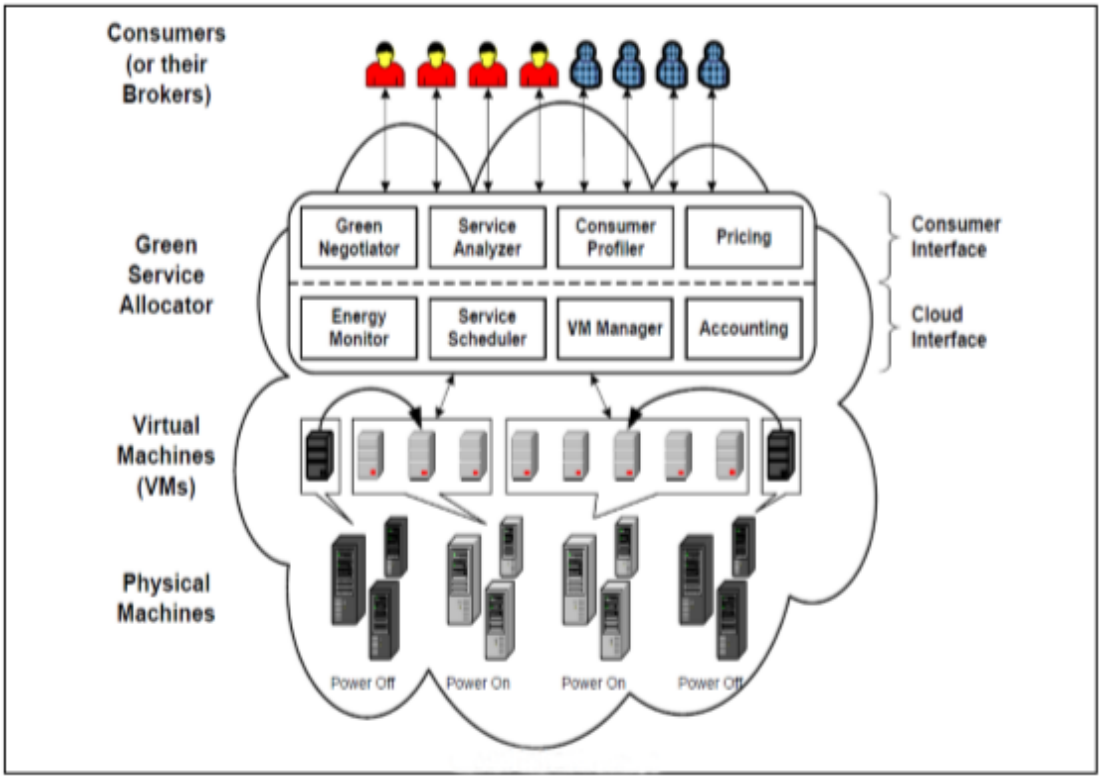


Figure 2.1: Cloud Computing Infrastructure

IV. Physical Machines

A hardware resource to meet the condition specification act as physical machine.

2.1.2. Cloud Computing Optimization

System load and system topology are the two major conception in load balancing. [17]. System load which depends on how the workload is loaded on the computer nodes. System load may follow centralized approach, distributed or hybrid approach. A single node able to manage the whole distribution in centralized approach. Each node is responsible to build its own load vector in distributed approach. This is done by collecting the applicable information by the other nodes. By checking the load vector, this approach then makes the decision. This technique is more resourceful in contrast with the centralized approach in cloud computing. Mixture of the

distributed and the centralized approach is the mixed approach. On the other hand which approach depends on the study of the information status of the nodes it's called system topology. System topology is promote classified into three approaches- static approach, dynamic approach and adaptive approach. On the design /architecture or on the implementation of the system Static approach depends on. Dynamic approach on the other hand depends on the decision making through load balancing to resourceful allocation the workload. The next is the adaptive approach which is most appropriate if the system status changes most repeatedly. This approach makes decisions by changing the limitation of the system when the state changes. Mainly three level of abstraction is there including Infrastructure in cloud computing 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. A cloud system has various data centers, which further includes various machines in cloud computing. Each machine then allocates the workload resourcefully to various virtual machines with the intention to save the energy [18]. The authentic implementation is done on the host machine but the client/user sees the resources on the virtual machine.

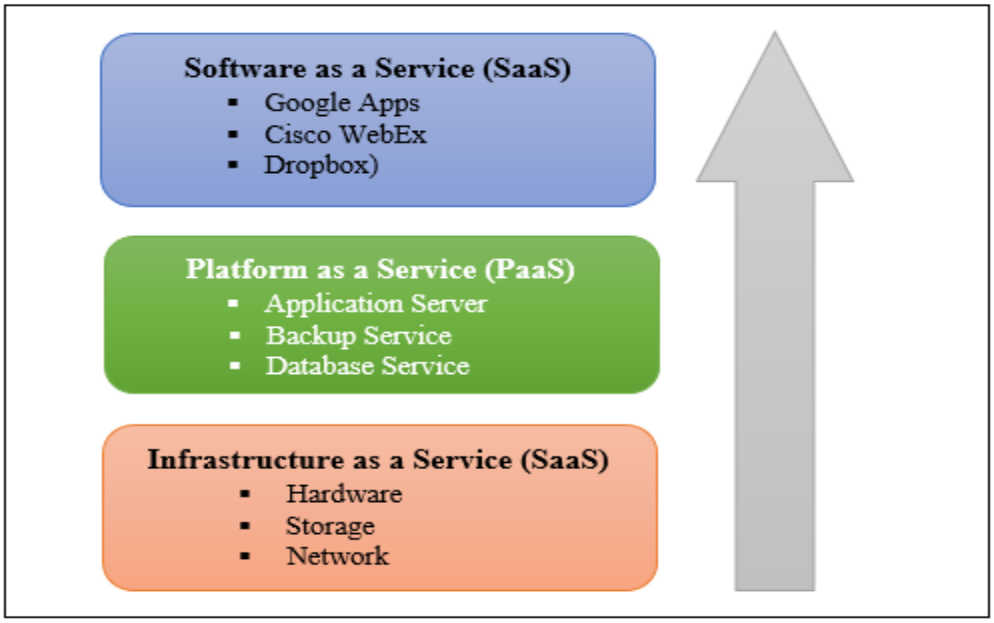


Figure 2.2: Three level abstraction in cloud computer

In cloud computing can work at two levels- local level and global level by the resource manager. At the local level is used to find the host machine of the local data center by the load balancer. At the global level is used to find the data center globally by the load balancer. After the resources are allocated to the cloud system the requirements may change. This may lead to the distort condition which needs optimization for better performance. In the several policies which are implemented to enlarge the throughput. The policies like Selection Policy first follows some selection criterion and picks which node needs to be migrated [19], Transfer Policy locates the mission to be performed at the time of the movement, and Location Policy depends on the selection of the lighter node to find the appropriate machine and the Information Policy that collect the information of the nodes.

2.2. Big Data and Cloud

The data is escalating massively in many forms like audio, text or video. It is not easy to handle that data with the conventional technologies. From the researchers Big Data has attracted huge attention. To handle that data which aids in large scale data processing by Big Data new emerging technology. Mapreduce divide its work and allocates its work to a variety of nodes or machines [20]. Another version of the Mapreduce is YARN which provides supplementary features [21]. When the work gets loaded into the slow running machine then the problem is arises, because run time of a machine depends on the multiple parameters including its processor speed. Three parts of map basically map reduce has, shuffle and the reduce part. Map task begin the mapping of the task. This step panel the task into subtasks. The reduce task groups the similar items. This can reduce in over loaded of the reduce tasks, as the map tasks executes in parallel. For the reason an algorithm was projected to reduce the overload [22]. The load balancing algorithm in the center of the map and the reduce tasks, able to separate the large tasks into smaller and then only send the smaller tasks to the reduce tasks and the entire procedure depends on the availability

2.3. Load Balancing Algorithms

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

2.3.1. Static Load Balancing Algorithms

The static load balancing algorithms are based on the former knowledge of the properties of the nodes and its capacity to process the new requests. The static load balancing algorithms also includes the processing power of the node. This only takes into deliberation the static properties of the node and is not flexible to any changes. A central load balancing decision model (CLBDM) which is a development to the round robin was proposed [23]. A connection time among the user and the node is calculated and is compared against the threshold value in this model. If the connection time surpass the threshold value then the connection will be completed. Then using the round robin approach the task will be assigned to the next node.

Processes are allocated to the processor in the round robin manner and this allocation is done locally in this approach. The main drawback of this approach is that as the processing time of the unusual processes are not the same so some nodes may remain redundant while the other nodes may busy. To handle the web traffic this approach is proper in the web servers where the https requests are distributed equally 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. Workload is assigned and may relocate to the nodes based on the computation in this approach. These algorithms also require continuous observing of the tasks. This is also more exact approach of load balancing in evaluation with the static approach.

2.4. Related Work

Shu Ching Wang et al. proposed Load Balancing Min-Min (LBMM) algorithm whose context consists of three levels [24]. First one is the request manager, second one is the service manager and the third one is the subdivision. The request manager's work is to receive the workload and then disperses it to the service manager. After the service manager receives the request, it split up the workload to increase the processing speed. Also based on some parameters including the CPU remaining memory, node availability and others, the service manager may assign the load to the service node. However as the process consists of three level it is slower than the other algorithms. Min-Min algorithm first checks which tasks need the minimum completion time and then the task with minimum time is assigned to the node, after that it updates new time by integrating the previous processing time of the machine with the time of the new task assigned. Then the assigned task is removed from the list. But main problem of this algorithm is that it may lead to starvation. Junjie Ni et al. proposed Load balancing algorithm for virtual machines [25] where central scheduling is used to calculate the resources available for the task. The available resource is then assigned for the processing. Details of the available resources is calculated by the Resource Monitor. This algorithm is based on the machine mapping. This process contains of several stages which includes request acceptance, collecting resource information and the resource availability calculated by controller to process the tasks. The task is given to the resource with the highest score. Then the application will be accessed by the client. The drawback of using this algorithm is that node capabilities and the network load are not take into account. Also the whole system will be shut down if a single point fails. Kumar Nishant et al. proposed an algorithm which is lot like an Ant's Behavior. This Algorithm is very similar to ant"s behavior as the ants march towards the region of large amount of resources [26]. An ant always looks for new food and uses that food sources to deliver back to the nest. First a head node is selected in this algorithm. This head node is selected based on the node with the large number of the neighboring nodes. As we know ant's always moves in one direction at a time in search for the food and after getting the food it heads back towards its nest following the same direction, just like that this algorithm proposed that the ant moves towards forward direction

finds the under loaded or overloaded node. If the ant encounter an under loaded node then the work is distributed across the under loaded nodes & if it 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. So for resource utilization this is an efficient technique. But the main limitation is that network might become overcrowded due to large amount of ants the network. Also status of the node after the ant's visit is also not taken into account. Che-Lun Hung et al. proposed Load Balancing Max-Min-Max Algorithm. This algorithm consists of two phase which is a combination of Load Balance Min-Min (LBMM) and Opportunistic Load Balancing (OLB) scheduling algorithms [27]. In LBMM technique, nodes information is available in advance, thus it is also known as static algorithm. In LBMM technique, execution time is reduced. In OLB (Opportunistic load balancing), a task is assigned to the node randomly. Each and every node of the process is kept in the working state. Thus load balancing is achieved. The combined approach of both LBMM and OLB is to maintain load balancing. In this combined job, first step is to check the completion time of each jon and then calculate the average time of the job. Second step is to choose the job with maximum response time. Third step is to select the unoccupied node with the minimum response time assigned the task to it. If the nodes are not available then again compute the result by checking the available and unavailable nodes. The above steps are repeated until all the tasks are executed. Martin Randles et al. proposed Honeybee Foraging Algorithm [28] which is based on the behavior of the honeybees. In this honeybee foraging algorithm, several servers are gathered together as a virtual server. Here a forager selects a virtual server randomly. Then each server maintains a queue for processing the request. After the processing, the server calculates the total profit. Depends upon the profit contribution, the request is processed by the server. If the computed profit is less, then the server returns to their forage. This maintains the balance of the load of the system. But also this computation of the profit can make an additional overhead which causes overall decrease in the throughput. Based on the dynamic and random sampling of the nodes Biased Random Sampling balances the load of the system. In this technique, virtual graph constructed to represents the connected nodes determines the load of the system. Free resources in the system are determined by the in-degree and representation of the node is constructed by the vertex. So to balance the workload, allocation of the work depends on in- degree of the node. When the allocation of tasks

are done, the in-degree is decremented by the value one, which means the availability of the free resource has decreased. After executing of the request, the node created an edge and the value of

the in-degree is incremented by the value one which means that the availability of the free resource has increased and this whole process is completed by random sampling [29]. The process is computed by the comparing parameter which is also known as the threshold parameter. Here the threshold value is denoted as maximum walk length and walk length is calculated by calculating the traversing of one node to another until it reaches a destination. After receiving the request, the load balancer selects a random node, and walk length of the current node is compared with the threshold level. If the walk length is higher than the threshold value then the task is assigned to that node and if it is lower than the job is assigned to the next node and the current walk length is incremented by one. Next node is considered as the neighbor of the current node. This is how this algorithm balances the workflow, but the calculation of the walk length creates an additional overhead.

In cloud computing, Active clustering [30] is a method to balance the load where its main aim is to group together the similar nodes. This grouped node then act as a cluster and the algorithm works on these groups. The construction of the cluster depends on the match maker node. In this technique, a node chooses the neighbor node of the different type which is known as the matchmaker node. This matchmaker node then connects with the neighbor node of the similar type as the initial node and then the matchmaker node is disconnected. Iterative approach is followed by this algorithm.

Yi Lua et al. proposed Join-Idle-Queue algorithm [31] which is used to balance the load of distributed data of large scale. This technique load balances the idle processors and its main concern is the availability of the idle processors across the dispatchers. The main aim of this algorithm is to decouple the lightly loaded processors from the task. Reducing the average queue length is the main advantage of this technique. The other advantages are less communication overhead and reducing load in the system. But the main drawback of this technique is that it is not scalable that's why this technique can't be used for modern web services which are dynamic in nature. Join idle queue depends on the behavior of two level system's. First level is dispatcher behavior where dispatcher first receives a work from the client. Then it checks if there are servers in the queue, if is finds any server then the job is allocated to it or else it selects a random

server. This is known as primary load balancing. Second level is the server's behavior which means after the completion of the jobs, idle processors are balanced across the dispatchers.

Jeffrey M. Galloway et al. proposed Power Aware Load Balancing (PALB) Algorithm [32]. This algorithm was designed with the aim to give cluster controller the control of computation. PALB has three main sections, which are balancing, upscale and the downscale section. Balancing section checks virtual machines state. If all the active nodes which are computed are utilizing more than 75%, then the least load one is distributed to the new virtual machine or else the load is distributed to the most utilized node to balance the workload. This threshold level of 75% is chosen. Upscale section powers up the additional nodes if utilization of the current node is more than 75%. Downscale sections purpose is to save the power cost by power down the underutilized nodes. So idles nodes are signaled to shut down by PALB and this shutdown signal is given to the nodes with less than 25% utilization. This algorithm mainly checks and decides which virtual machine should be instantiated and also decides which nodes should be operated.

In the cloud computing, Equally Spread Active Execution (ESAE) algorithm [33] further improved the problem of random load. In this algorithm as the tasks are submitted, they are queued. If the task size and the size of the virtual machine matches then the job is assigned by the job scheduler based on the priority. This way ESAE algorithm improves the response time and due to equal distribution of the job, overall cost is reduced. Tin-Yu Wu et al. proposed an algorithm called Index Name Server (INS) whose purpose to minimize the data duplication [34]. In this algorithm it calculates an optimal selection. The selection depends on several different parameters which are maximum bandwidth, position of the server, weight factor, hash code of the data etc. The other main parameter is to check if additional nodes can be handled by the connection or not means the busy level. The busy level if it will classified into three parts- First part is cannot able to handle more incoming nodes and the connection is busy. Second part is additional nodes can be added and the connection is not busy. Third is the connection is limited. The limitation of this method is that it does not predict the future performance of the nodes. Rich Lee et al. proposed Weighted Least Connection (WLC) [35] to find out the node with the minimum number of 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 bandwidth and processing speed etc. This limitation is

improved by an algorithm called Exponential Smooth Forecast based on Weighted Least Connection (ESWLC). ESWLC takes into consideration the node capabilities and the time series

[36]. This is done with the help of making decision on the basis of number of connections, memory, CPU power etc. The selection of the node is created on the exponential smoothing. Al-Jaroodi et al. proposed Dual Direction FTP algorithm [37]. It works by splitting the file of size "n" into "n/2" partitions. Task is allocated to the nodes and processing is done in terms of blocks. The nodes are work independently. Example a node starts processing form the first block and continues its processing incrementally though the another node starts the processing from the last node and it does its processing decrementally. All the nodes will end up processing the whole system in this manner. This is reduce the all response time. This algorithm also reduces the network overhead as it reduces the communication between the nodes and the client. Therefore network load is also taken into consideration in this algorithm. The main limitation of this approach is that all replication of the files is needed; therefore high memory is essential for the nodes. M. Sharma et al. proposed Throttled algorithm [38] to assign the workload on the virtual machines for effective resource utilization. But the resource utilization and response time was improved in 2013 and the changed Throttled algorithm was proposed [39]. The modified algorithm presented the method and it maintains the index of the virtual machines. The user firstly requested that the load balancer to find the appropriate virtual machine. For example 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 is not priority of the work. A method was proposed [40] to consider the important 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 smallest. In 2014, an algorithm was proposed [41] to distribute the workload on the least loaded virtual machines and the purpose to improve the resource utilization. This algorithm helps to find out allocate the work accordingly and the least loaded virtual machine. Suppose there are "p" numbers of users and "q" virtual machines are there for processing. The algorithm works by initial passing the request to the load balancer. The load balancer keeps the index of virtual machines. From starting, all the virtual machines are free, so they follow the round robin algorithm initially but when 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 instantly granted to it. If it is not available at the moment means if it is already

given 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 Balance load balancing And Dynamic Compare algorithm (DCABA) to improve the cloud [42]. If the load of current selected node is bigger than the other selected node, it transfers the part of the load which is extra to the another node. In the proposed procedure, by using compare and equilibrium condition, balance strategy is maintained. So, the resource utilization is monitored by using process migration. 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. This algorithm is divided into the two sections. By checking the current load, we decide which part of the section is to be traversed. Therefore 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. It is support the idea of the green computing. Both the sections are further subdivided. Section one determines a condition when the load is more than the threshold value. So this is the condition of overloading. This is again resolved by using load balancing technique to balance the extra load to the other host. Then we search the machine with the minimum probability of the load so the load can be distributed to that machine. Section two determines the condition when the load is below the limit value. This condition's name is under loading. This condition increases the total 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. So the main aim is to search the other machine where the load is transferred. Gulshan Soni et al. proposed Central Load Balancer algorithm in 2014 [43]. This algorithm distributed the workload between virtual machines and is based on the computing capabilities and hardware configuration and. This is better and reliable resource utilization is achieved through this load balancing algorithm. Figure 2.3 shows the architecture of the Central Load Balancer.

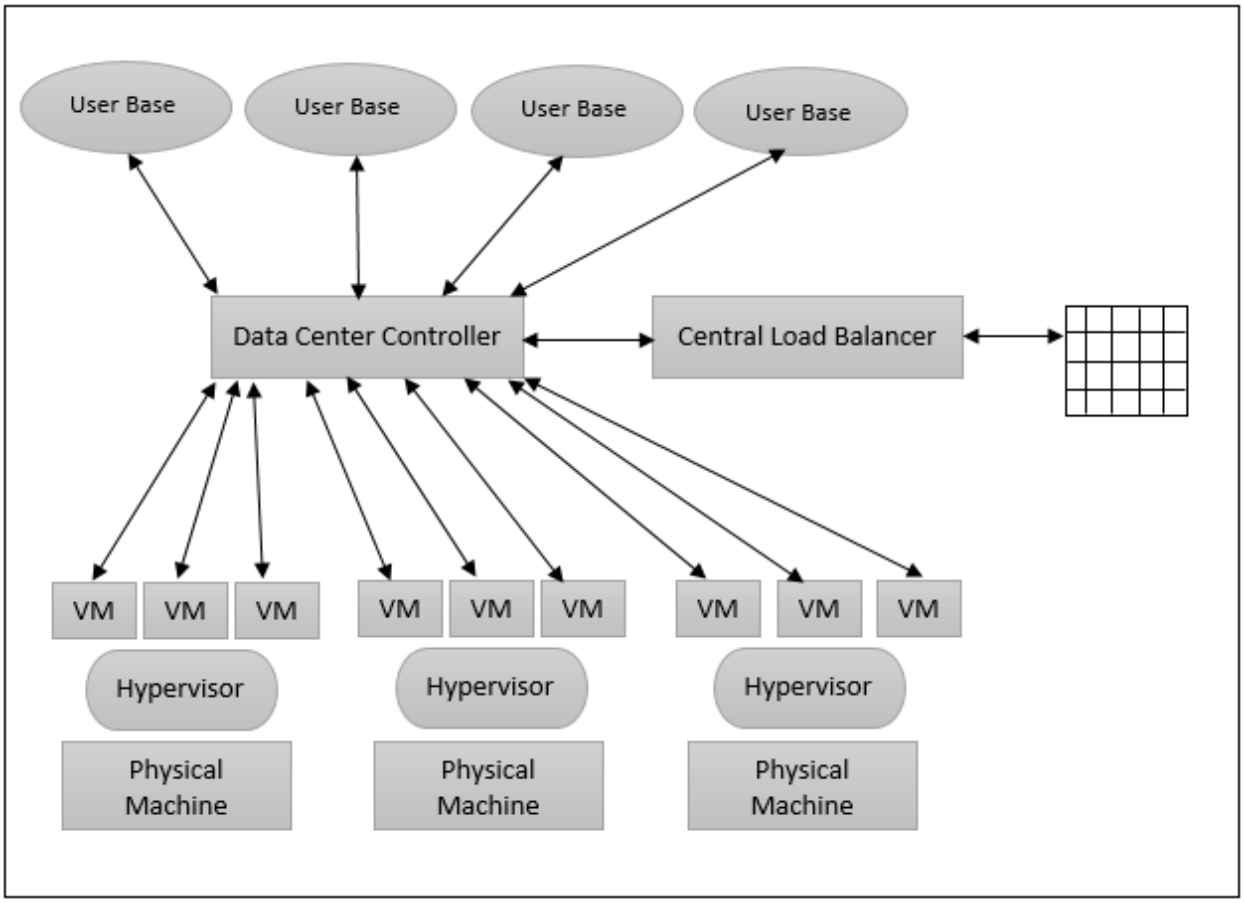


Figure 2.3: Architecture of Central Load Balancer Flowchart

In this algorithm, first the request arrives at the Data Center. Data Center before asks the central load balancer to allocate the requests. The flowchart of the algorithm has shown in Figure 2.4. It keeps the table that contains the virtual machine ID, the set priorities and states 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 appropriately. If the virtual machine state is free, then the free virtual machine ID is returned to the Central Load Balancer.

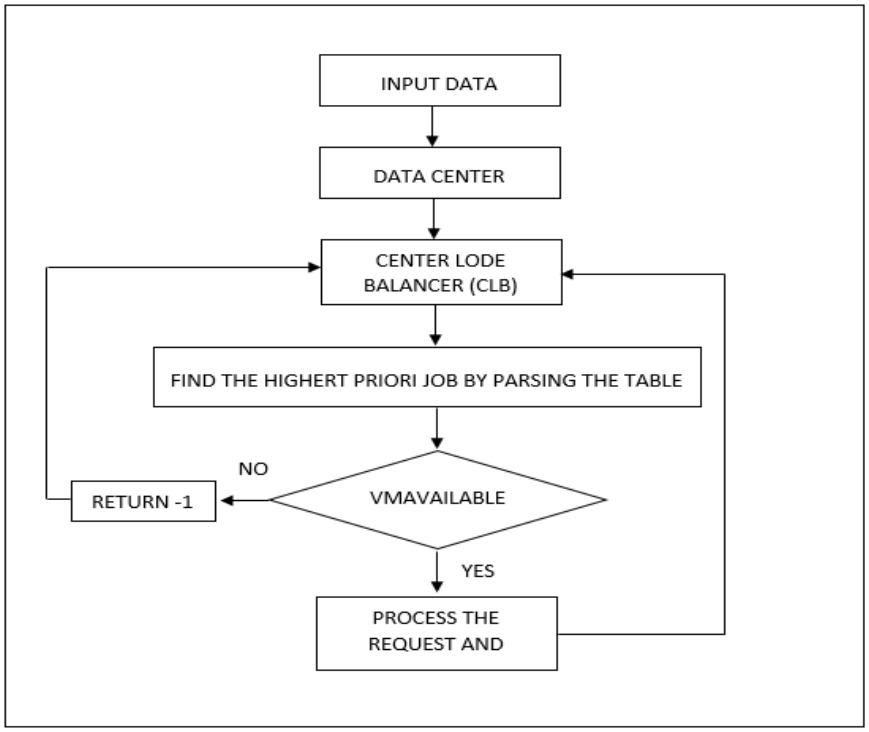


Figure 2.4: Central Load Balancer Flowchart

Load Balancer is responsible for the priority computation. Table will be updated when complete load to the virtual machine. Data Center gives the notification to Load Balancer controller of the new allocation. If the virtual machine is busy, then the ID -one is returned and the request is queued in the data center organizer. Central Load Balancer is linked with the users and the virtual machines. It is used to analyze the priorities and then allocates the work based on the processing speed of the machine. Priority has two factors including processor speed and the memory of the machine. After the virtual machine finishes the request, Data center gives the notification to the Central Load Balancer of the completion of the request. Now the data center 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 over loading and the under loading. This algorithm competently 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 least loaded virtual machines the workload on [44]. It helps to improve the resource utilization. The entire process has shown in Figure 2.5.

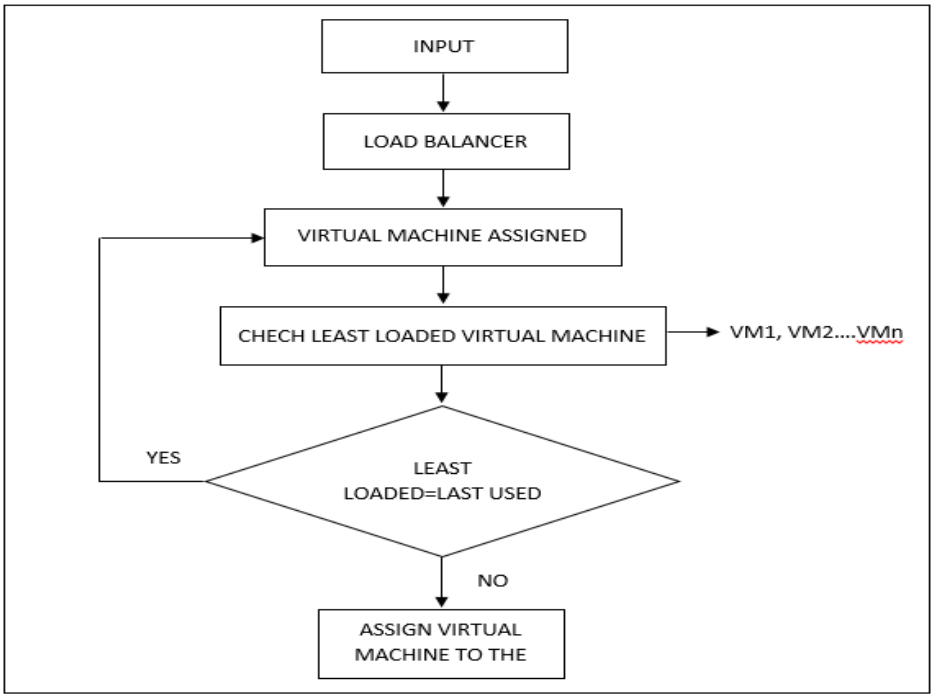


Figure 2.5: VM Assign Load Balancing Flowchart

The least loaded virtual machine and allocate, this algorithm helps to find out the least loaded virtual machine and allocate the work consequently. The algorithm works by first passing the request to the load balancer.

The load balancer maintains the index of virtual machines. At start, they follow the round robin algorithm initially all the virtual machines are free so but as the next request comes, the algorithm checks the virtual machine table. If the requested virtual machine is unfilled and is not yet assigned, then the requested virtual machine is directly granted to it. If it is already assigned to do other work if it is not available at the moment means, 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 all other nodes in around robin manner the first node randomly and then distributes jobs.	Some nodes might be heavily loaded and some are not. Since the running time of any process is not known past to execution, there is a possibility that nodes may get deeply loaded.
Weighted Round Robin	In this algorithm, the requests are processed, a weight is assigned to each node and depends on the weight.	As defined prediction of execution time is not possible, therefore this algorithm is not preferred.
Map reduce Based entity Determination [23]	Allocate the entities of large blocks among various 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.	If the connection goes above the threshold level problem arises, Since this method works based on the round robin performance
Ant Colony Optimization [25]	.Under loaded node is found at the opening of the search . It is Distributed, so there is no single point of failure. . Ants can collect the information faster.	.Due to large number of ants, network may be blocked. . The position of the nodes after the ant's visit is not taken into consideration.
Load Balancing Min-Min (LBMM) [26]	Job with the lowest time is effected first.	The problem of this algorithm is that some jobs may experience starvation.

Load Balancing Max-Min- Max OLB + LBMM [28]	Uses resourceful Load Balancing (OLB) to keep each node busy and uses Load Balancing Min-Min (LBMM) to achieve minimum execution of each task.	Answer time can be improved
Performance [29]	Achieves global Load Balancing through local server and used for large balance cloud systems	The computation of the profit may cause an extra overhead which results in overall Decrease in the material.
Biased Random Sampling [30]	Succeeds Load Balancing across all system nodes using random sampling of system domain	The walk length creates an additional overhead.
Active Grouping [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 contributors for the availability of the idle processors at each transmitter. 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 consistency.
Power Aware Load Balancing (PALB) [33]	PALB algorithm was designed with the resolve to give computation control to the group controller	Overloaded, under loaded case is not reflected.
Equally Spread Active Execution (ESAE) algorithm [34]	In the ESAE algorithm, as the tasks are submitted, the job is assigned, they are queued. If the task size and the size of the VM match. This is done by the job scheduler based on the priority.	Power consumption case Is not considered.
Weighted Least Connection (WLC) [36]	It finds out the node with the least number of the connections. If the task is assigned to it, the node with the least number of	It does not takes into reflection important factors including the processing speed,

	connections is found.	bandwidth etc.
Exponential Smooth Forecast based on Weighted Least Connection (ESWLC) [37]	. More Perfect results than WLC . Takes into consideration two main factors, node capabilities and time Series.	. Complex . Prediction algorithm. requires existing data and has long processing time

Dual Direction FTP (DDFTP) [38]	DDFTP is as Wild 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 VM for effective resource operation	Failed to distribute load regularly, overloading initial VMs and leaving others underutilized
Modified Throttled Algorithm [40]	Developed resource Utilization as compared with the Checked algorithm.	This process does not taken into consideration the priority of the work.
Optimal Virtual Machine Allocate Load Balancing [42]	. This algorithm helps to find out the least loaded virtual machine and allocate the work therefore. . The load balancer maintains the index of virtual machines.	The proposed algorithm can static be developed by taking some more dynamic positions 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 planned technique, by using compare and balance strategy, balance condition is maintained.	Cloud application service running on any cloud in a datacenter may change their resource requirement.

Central Load Balancer [44]	This algorithm powerfully distributes the workload among virtual machines, based on hardware structure and the computing capabilities. The load is balanced based on the basis of priority calculated considering hardware parameters.	It has reflected memory and the speed as the parameters for priority calculation. Other factors may also take into consideration to improve the operation.

Table 4: Comparison no various Load Balancing Algorithms

Chapter III

PROBLEM STATEMENT

Chapter 3

PROBLEM STATEMENT

Problem Definition- There are few issues in cloud computing and one of them is Overloading because of the random arrival of the tasks. Due to random utilization of the CPU, some resources remain idle whereas some are heavily loaded. Another one is Power Consumption Issues on Data Centers as most of the power in cloud computing is wasted because of underutilization and ideality of resources at data centers. In our approach, we will be using Load Balancing to distribute the load over the network across a large number of virtual machines or CPU which will maximizes the performance and minimizes the response time. Also we will be using live migration of VMs so that we can separate hardware and software and it will also brief us about fault management, load balancing and low-level system maintenance.

3.1. Research gap

Throttled Load Balancing algorithm works only with pre-defined number of the tasks which are allocated. But the problem occurs when more than the pre-defined number of the tasks arrives. In Round Robin we face similar kind of problem where requests are queued till the VM becomes available. Also in active monitoring load balancing algorithm, the work is allocated by distributing the entire workload equally to the available virtual machines. But limitation of these approaches is that if the hardware configurations of one machine is different than other then it would be a problem to balance the load. Although Central Load Balancer spread the work to the virtual machines according to the priority and the availability of the machines. But still current utilization of the nodes are not taken into account.

So our main aim is to develop an approach that will be dynamic and avoids the situation of over loading and under loading. We will be using simulator known as CloudSim and its toolkit to execute our proposed work. Our approach will not only use load balancing algorithm based on

resource utilization but also it will analyses various Load Balancing Algorithms which will help us to balance the task in an efficient way.

Also we will use Live Migration to save power while balancing under loaded and overloaded VM's. An under loaded VM which underutilize its CPU capacity are migrated to those nodes whose residual capacity is big enough to hold them. So the last node is switched off to save power. Whereas an overloaded VM will cross utilization capacity so we will migrate it to under loaded VM's [7, 8, 9]. Using Live Migration can lead to the performance degradation of the node. So continuous monitoring approach can be applied to lessen the VM migration and ensure quality of service.

Chapter IV

PROBLEM FORMULATION

Chapter 4

PROBLEM Formulation

To avoid avoids overloading and underloading problem and to save power as well an efficient Load balancing & Live Migration algorithm are combined and used as a single algorithm. This algorithm is connected to the datacenter and all its users. All the virtual machines are managed by the datacenter controller. Priorities of the virtual machines is computed by parsing its table, based on their speed, memory and power consumption. It will then pass the load to highest priority VM. Also this algorithm will use Live Migration to save power while balancing underloaded and overloaded VM's.

4.1. System Architecture

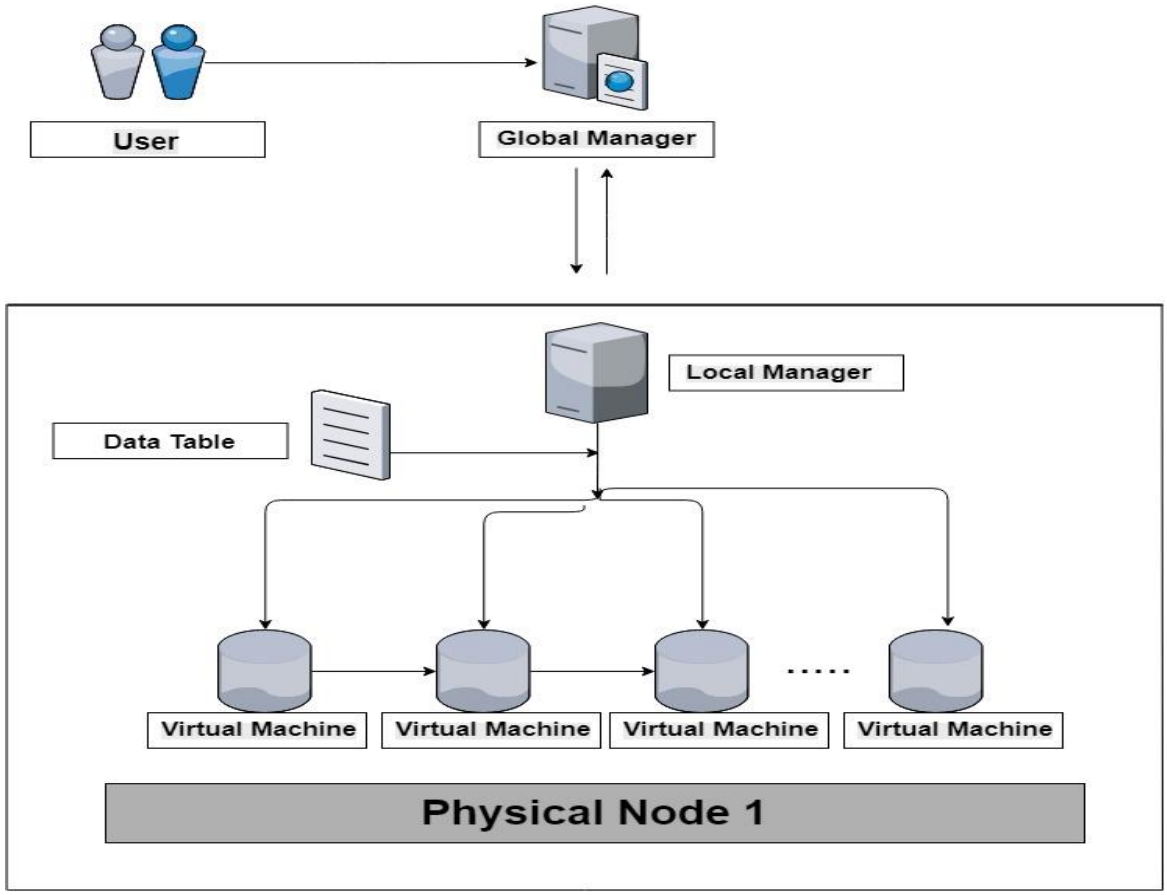


Figure 4.1: System Architecture

This System Model consists of Global Manger, Local Manager & Virtual Machines. First a User sends request to the Global Manager which handles the request and sends data to The Local Manager. Then the Local Manager manages a data table which contains all the Virtual Machines. This table evaluate all the Virtual Machines and makes a priority list in a order of most to least priority and then assign tasks to the Virtual Machines accordingly. Also this local manager continuously monitor the Virtual Machines so that if there is any overloading to any virtual machine then the tasks will be migrated to any idle virtual machine. Later the local manager reports the global manager about the utilization check of its node. And thus, the global manager keeps the check of overall utilization of the resource.

4.2. Power V/S Utilization Calculation

Many studies [17, 18] shows the power consumption by servers can be described by a relation between the power consumption and CPU utilization. These studies as say that an average power consumed by an idle server is 70% of power consumed by fully utilized server so, we considered the power consumption as CPU utilization $C(0)$ by as shows in (1):

$C(0) = C_{max}(0.7 + 0.3\ O)$ ... ... (1)

Where C_{max} 250w is for modern calculation server and u is the CPU utilization [20].But CPU utilization change with respect to time i.e. O(t).So total energy consumed (F) as shown in (2):

$F = \int tC(O(t))\, dt$ ... ... (2)

So the total energy consumption can be measured from CPU utilization from this model.

4.3. Price Lead by Relocation

We propose decrease in power consumption using live relocation which results in reducing operation cost for the data center. We consider here cost as shown in (3):

$P_{total} = P * F$ ... ... (3)

When P is the cost of 1kW Power. We would also like to show comparison of costs using wirh and without relocation.

4.4. SLA Violation Calculation

QoS needed to be met for Cloud Computing environments. QoS is determined in the form of SLA (Service Level Agreement), which is determined either by minimum throughput or maximizes response time. This can differ from system to system. For our studies, we consider SLA violation as shown in (4):

$$SLA = \frac{\sum(requested\ MIPS\ ) - \sum(allocated\ MIPS)}{\sum(reuested\ MIPS)} \ldots\ldots\ldots\ldots (4)$$

The percentage of this value will show CPU is not allocated even if it ids demanded. Therefore, in order to increase the QoS for the end-users, our prior goal is to minimize this SLA from getting violated.

4.5. Proposed Scheme

Here, we proposed dynamic threshold based scheme. We divide the algorithm in two parts :(1) Selection of V_Mac for migration and (2) Placing the V_Mac on proper host.

4.6. Selecting V_Mac for Relocation

The selection of VM for migration is done to optimize the allocation. Here, we first calculated the CPU utilization of all V_Mac as shown below in (5):

$$O\ v\_Mac = \frac{total\ Requent\ MIPS}{total\ MIPS\ for\ that\ V_{Mac}} \ldots\ldots\ldots\ldots (5)$$

And hence then in our scheme we considered two threshold value:

a) Higher threshold value when the utilization is above this value. The CPU will be considered overloaded so we migrate dome of the V_Mac. Here 7, so went on calculating this value i.e. higher for each host separately by following equations in (6):

$Sum = \sum$O v_Mac

$SQ\_Rcp = \sqrt{(\sum \text{O v\_Mac})^2}$

T_ higher =1-(((Chl *Sq_ rcp) +sum)-((Cll *Sqr) +sum))……. (6)

Where for each host we preserve amount of CPU capacity by higher (chl) and leadt (cll) probability limits

b)Least threshold value the node is considered to be underutilized when the CPU utilization is below this value so all V_Mac are relocation to other node. From our study in [13], we considered that if the CPU utilization is above 30%, lower threshold (T_least) is always 0.3. So, we define equations for calculating least threshold for each node as follows in (7):

$Sum = \sum$O v_mac / n

$Sq\_rcp = \sqrt{(\sum 0\text{v\_mac} - \text{sum})^2}$

T_lest = sum - (P1 * sq_rcp), if CPU utilization is < 30%

=0.3 …….. (7) , if CPU utilization is >= 30%

Where, we considered P1 as probability limit of least threshold and n is number V_Mac on the host. After defining the dynamicity of lower and higher threshold from the equation (7) and (6) respectively, we consider our theory for Dynamic Threshold based live relocation as shown in the Algorithm 1.

4.7.An Efficient Load Balancing based on Resource Utilization

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

1. Begin
2. Request from user and data sender handler
3. Local Manager maintain parity table
4. If $P > T$
5. Send those VM's
6. Best fit values consumption for any VM's
7. If $t > BFCVM$
8. Then add each that VM and remove current host
9. Set those VM's to those host machine
10. Go to step 18
11. Else
12. Receipt VM
13. Go to step 18
14. End if
15. Else
16. Send the request
17. End if
18. Update table

4.8. Activity Diagram of Load Balancer

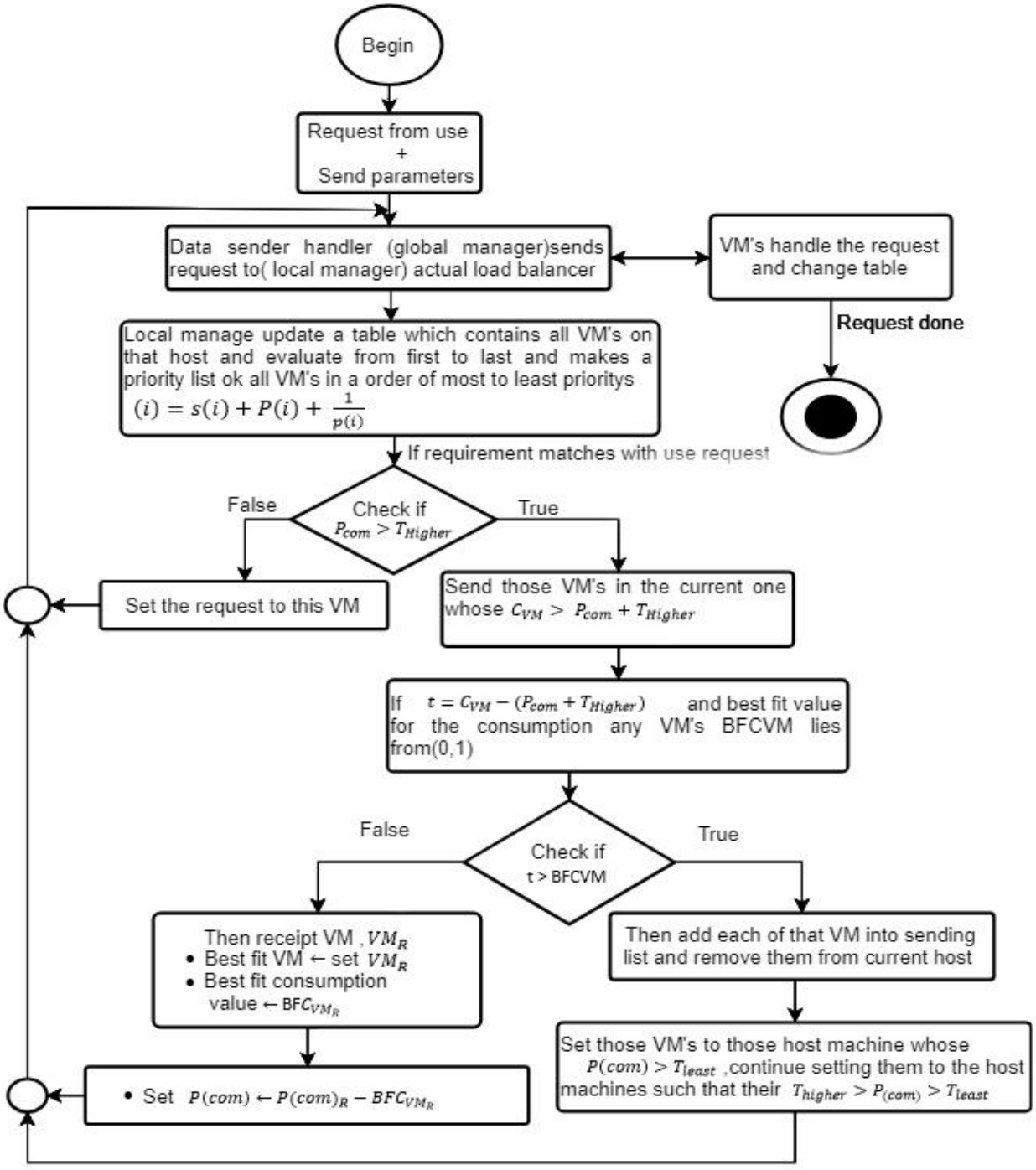


Figure 4.2: Activity Diagram

4.9. Power Utilization Model

CPU utilize the main energy. Power utilization can be calculated by considering CPU consumption. Idle server consumes 75 percent of all the power, utilized by CPU running at the full speed. The power model states [45] that

P (cons) = f * P higher + (1 − f) * P higher * cons

Where,

P higher =Power Utilization at the maximum level (Maximum Utilized)

cons = CPU Consumption

f= the fragments of power consumed by the idle server

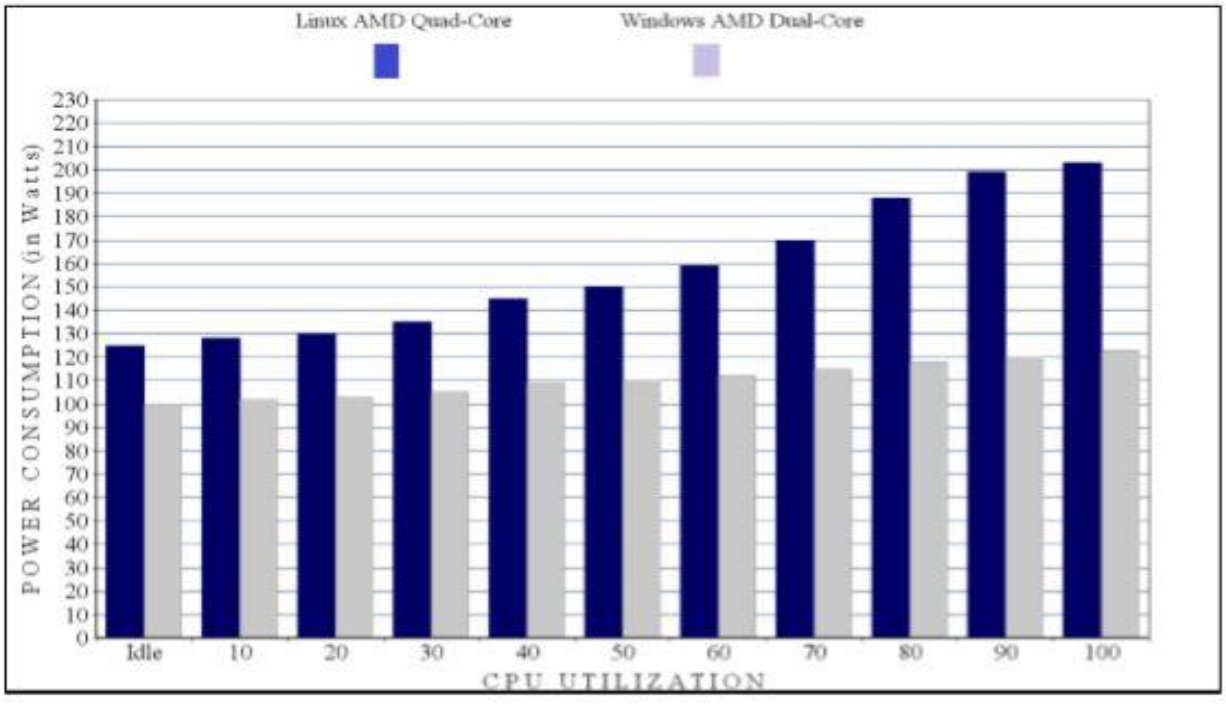


Figure 4.3. Power Utilization Model[45]

4.10. Cloud Simulation

More than 10.000 machines in less than 5 seconds with the help of only 75MB of RAM are instantiate by simulator called Cloudism[46]. Functionalities of the basic GridSim are extended by this simulator. And various research projects have extended this. This extended functionalities contains resource which supervise between the virtual machines, modeling storage and the application servers, simulation of the cloud etc. Many research fields uses CloudSim & this research fields include energy efficient cloud scheduling algorithms [47], and improvement in the communication flow includes the green computing energy scheduling algorithms [48]. The CloudSim is further updated to be more user friendly. The CloudSim is also used for the educational purposes which is known as TeachCloud [49] which uses the graphical interface so that the students become more interested and perform the experiments on the cloud easily. CloudSim setup an environment for load balancing algorithms which allows the virtual machines to be managed by the hosts. Later datacenters manages the hosts. A cloudsim contains four entities which manage the cloud resources. These four entities are Datacenters, Host, and Virtual Machines & Applications. Hosts are the pre-configured servers which got processing capabilities. Infrastructure services of the cloud are provided by the Datacenter and also it acts as a home to several hosts and the gathering of all the hosts lead to datacenter entity. Services to the cloud are provided by the Host. It usually have its own memory and the storage & Speed of the processing is defined by MIPS (Million Instructions per Second). Host also act as a home to several virtual machines and the gathering of all the virtual machines makes it a host entity. Virtual machines are mapped to the Host. Requirements such as storage requirement, processing requirement, memory requirement etc. are included in the matching characteristics. Therefore, the same instances of the hosts can be mapped to the same instances of the virtual machines based on the available scenarios. So Virtual machine execute the applications .Many services are provided CloudSim provides various services as shown in the Figure 4.3. The main elements of the CloudSim are also shown in the figure.

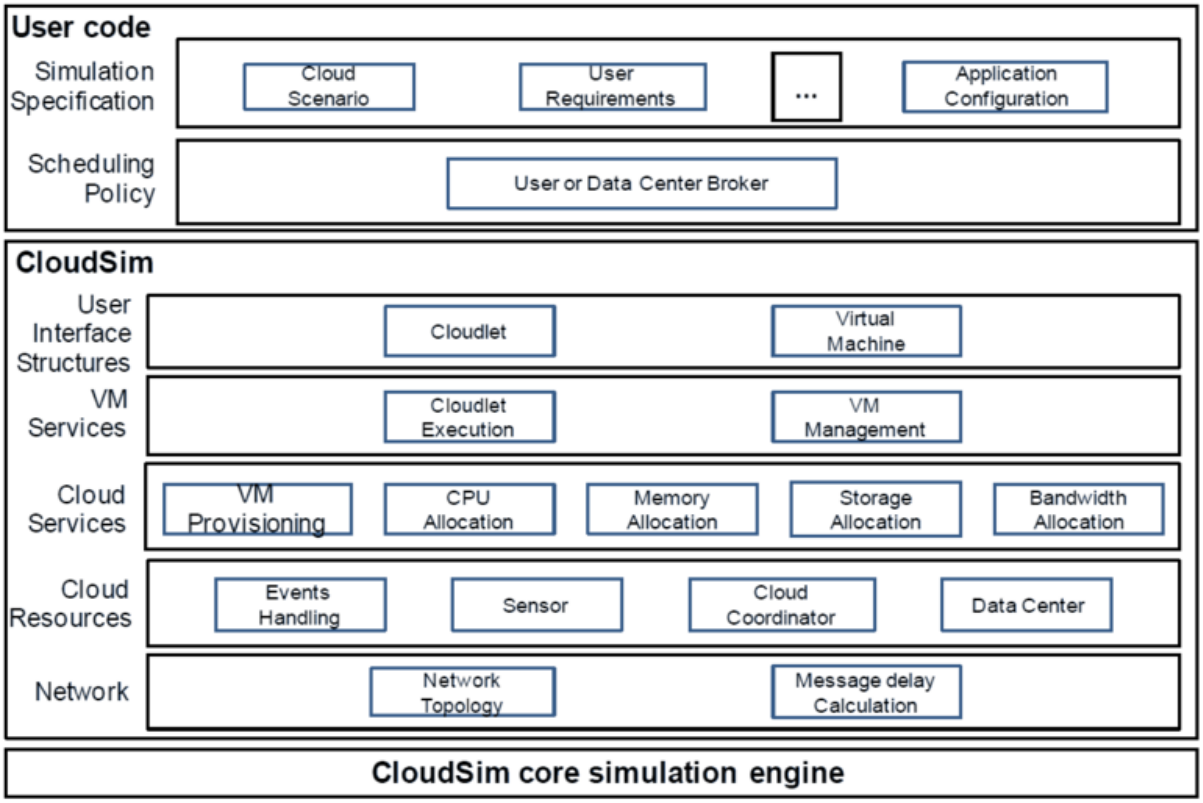


Figure 4.4: CloudSim Layered Architecture [50]

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 V

EXPERIMENT RESULT

Chapter 5
Experiment Result

We have implemented the proposed algorithm in the simulation software CloudSim and CloudAnalyst which is based on Java language. The simulation has considered three user bases (UB1, UB2, UB3) and three Data Center (DC1, DC2, DC3) each with multiple Virtual Machines (VM1, VM2, VM3,…). The software and hardware specifications are as follows:

5.1 Software and Hardware specifications:

Software specifications:

Windows-7 OS

Java software version 1.7

Eclipse IDE 8.0

ClodAnalyst

CloudSim 3.0

Hardware specifications:

X86 AMD Processor

3 GB RAM

2.67 Ghz Processor

Parameter specifications (used in CloudAnalyst for simulation):

Linux Operating System (OS)

Xen Virtual Machine Monitor (VMM)

204800 Mb Memory size

100000000 Mb storage capacity

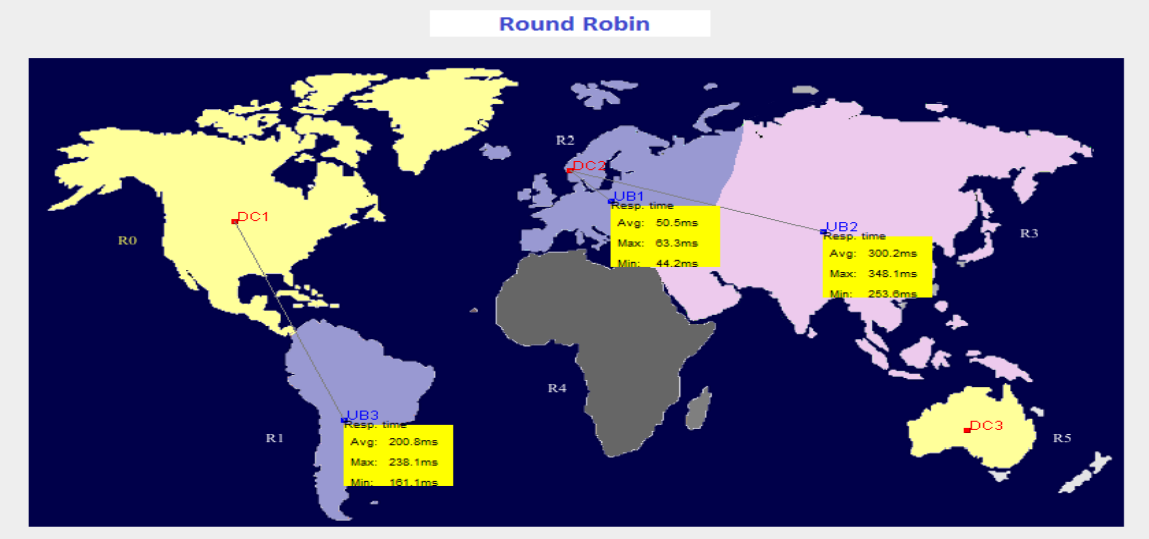
1000000 Network Bandwidth (BW)

Time-shared VM policy

5.2 Simulation Procedure

- At first we have added dynamic load balancing policy to CloudAnalysttoolkit via configuring it in CloudSim.
- Then we've configured the main configuration simulation criteria in Cloudanalyst by adding three user bases and their regions as let's say – 2, 4 and 5; then selected the duration of simulation for 10 minutes
- Next we've chosen the service broker policy as : Reconfigure dynamically with load balancing
- We've added three Data centers DC1, DC2 and DC3 with each DC having 5 VMs with 512 Mb RAM and Network BW 1000 kbps.
- In data center configuration, we've chosen their specifications as stated in section 5.1
- In advanced configuration, we've kept the user grouping factor in UBs (equivalent to number of simultaneous users from a single UB) and request grouping factor (equivalent to number of simultaneous requests that a single server instance can support) as 10.
- Then we've chosen load balancing policies one after another for each simulation across VMs in a single DC as respectively: Round Robin Policy, Equally Spread Current Execution load (ESCE) Policy, Throttled Policy and then our proposed Dynamic Load Balancing Policy.
- The results are shown in the sections below

5.2.1 Round Robin Policy



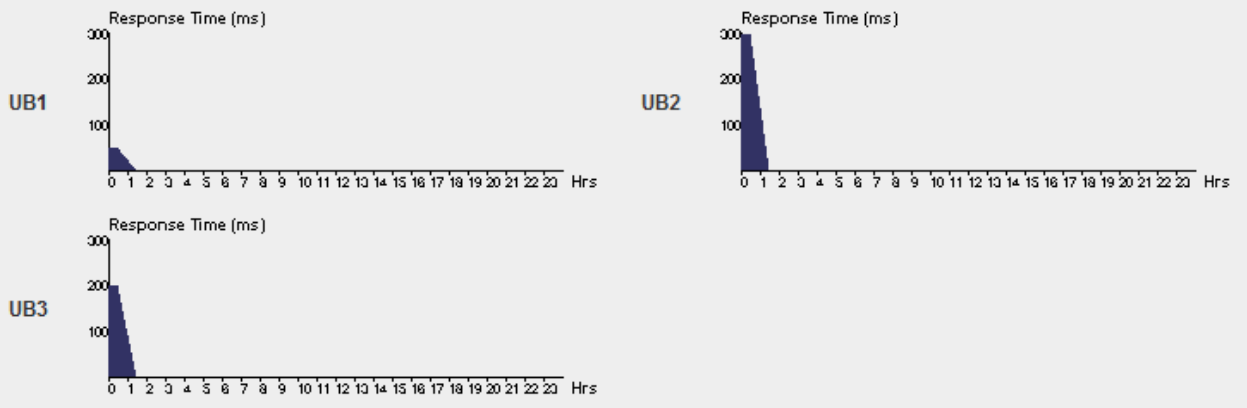
Overall Response Time Summary

	Avg (ms)	Min (ms)	Max (ms)
Overall response time:	181.73	44.23	348.14
Data Center processing time:	0.40	0.00	1.23

Response Time by Region

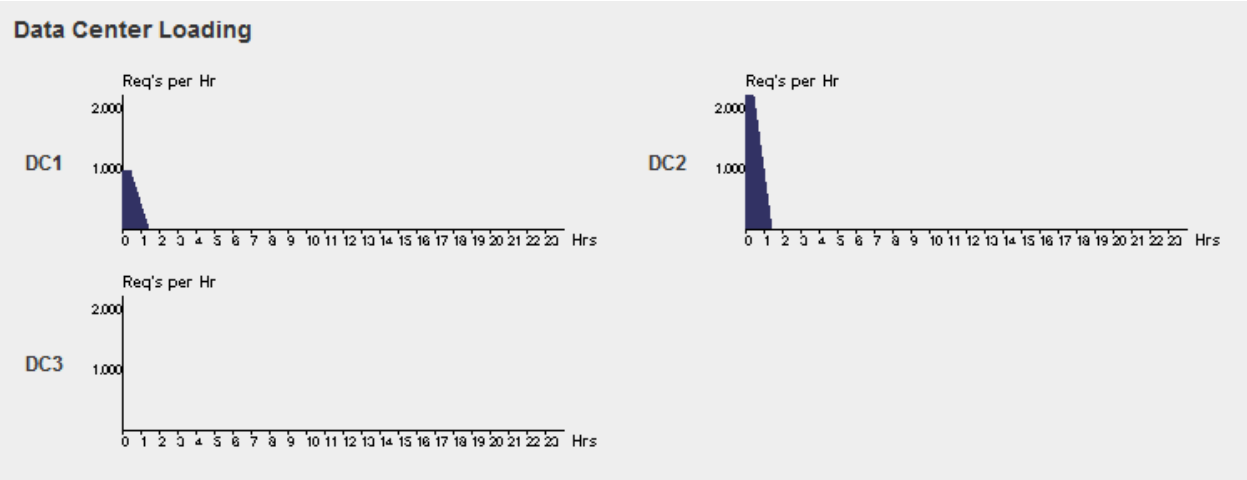
User base	Avg (ms)	Min (ms)	Max (ms)
UB1	50.52	44.23	63.31
UB2	300.17	253.63	348.14
UB3	200.80	161.11	238.12

User Base Hourly Response Times



Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	0.10	0.02	0.11
DC2	0.53	0.02	1.23
DC3	0.00	0.00	0.00

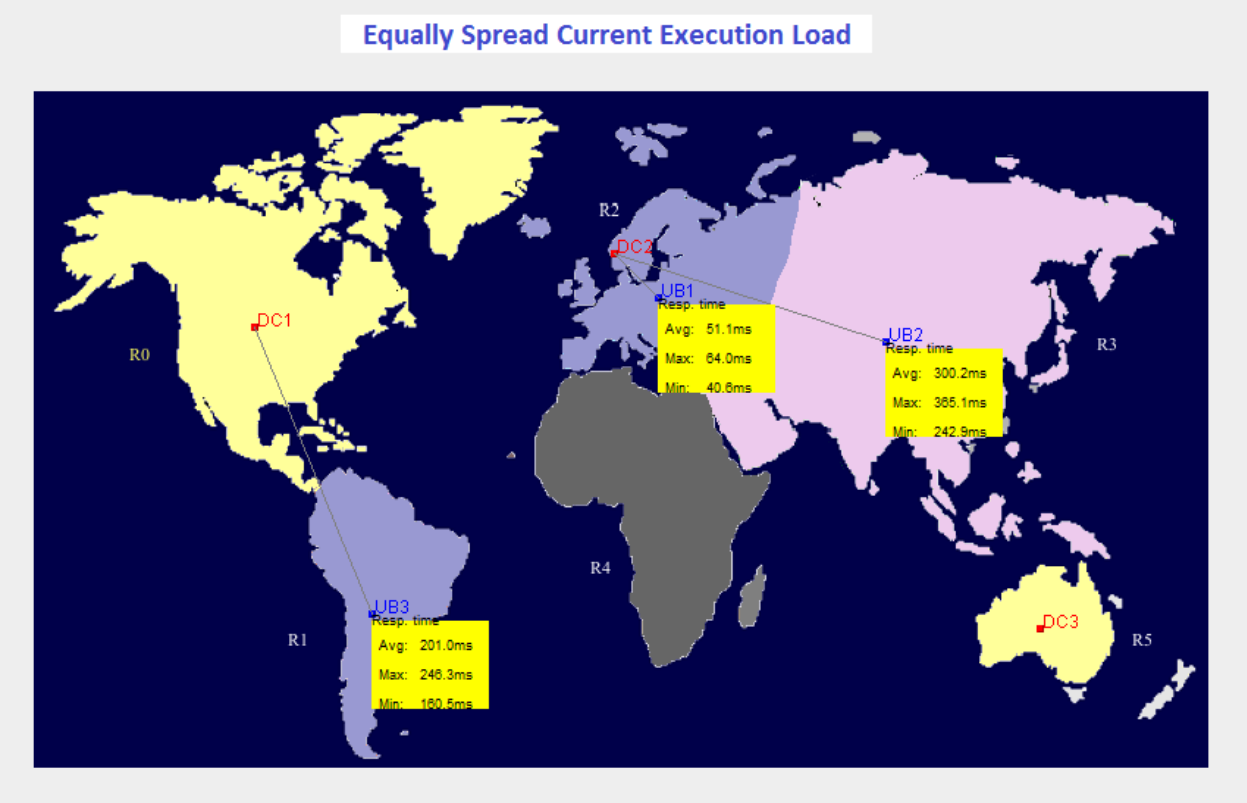


Cost

Total Virtual Machine Cost ($):	0.30
Total Data Transfer Cost ($):	0.03
Grand Total($) :	0.34

Data Center	**VM Cost $**	**Data Transfer Cost $**	**Total $**
DC1	0.14	0.02	0.17
DC2	0.09	0.01	0.10
DC3	0.07	0.00	0.07

5.2.2 Equally Spread Current Execution load (ESCE) Policy



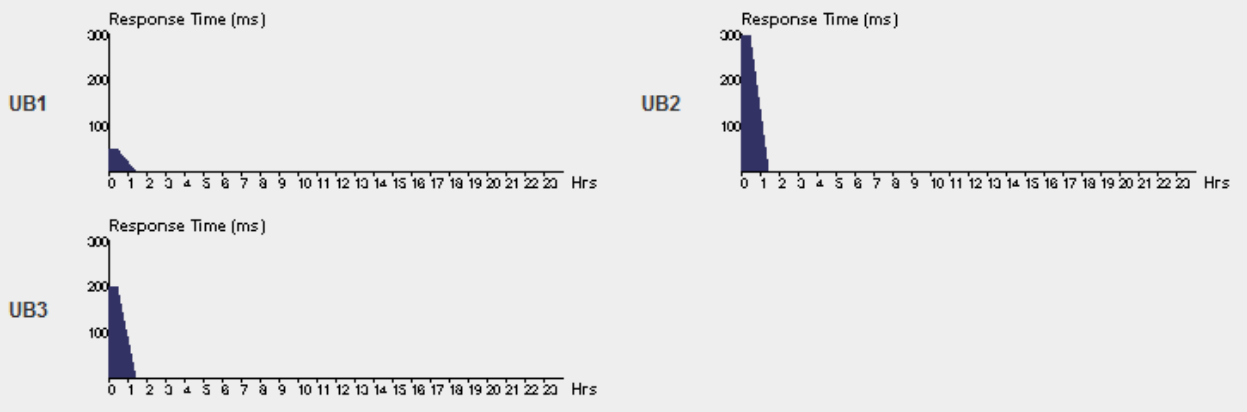
Overall Response Time Summary

	Avg (ms)	**Min (ms)**	**Max (ms)**
Overall response time:	184.82	40.63	365.12
Data Center processing time:	0.88	0.00	3.45

Response Time by Region

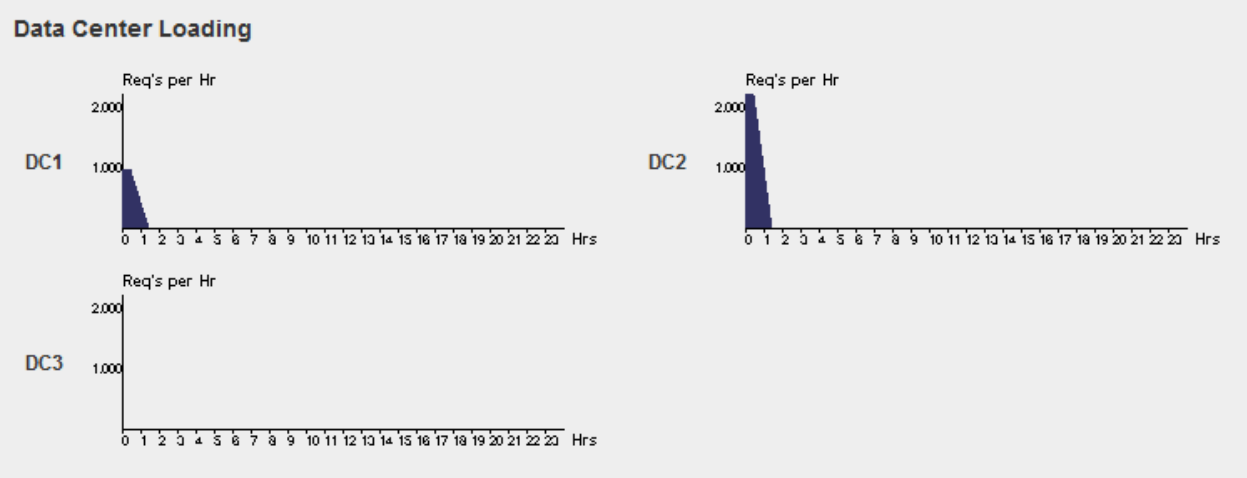
User base	Avg (ms)	Min (ms)	Max (ms)
UB1	51.15	40.63	63.98
UB2	300.18	242.89	365.12
UB3	200.99	160.47	246.28

User Base Hourly Response Times



Data Center Request Servicing Times

Data Center	Avg (ms)	Min (ms)	Max (ms)
DC1	0.34	0.02	1.40
DC2	1.14	0.02	3.45
DC3	0.00	0.00	0.00

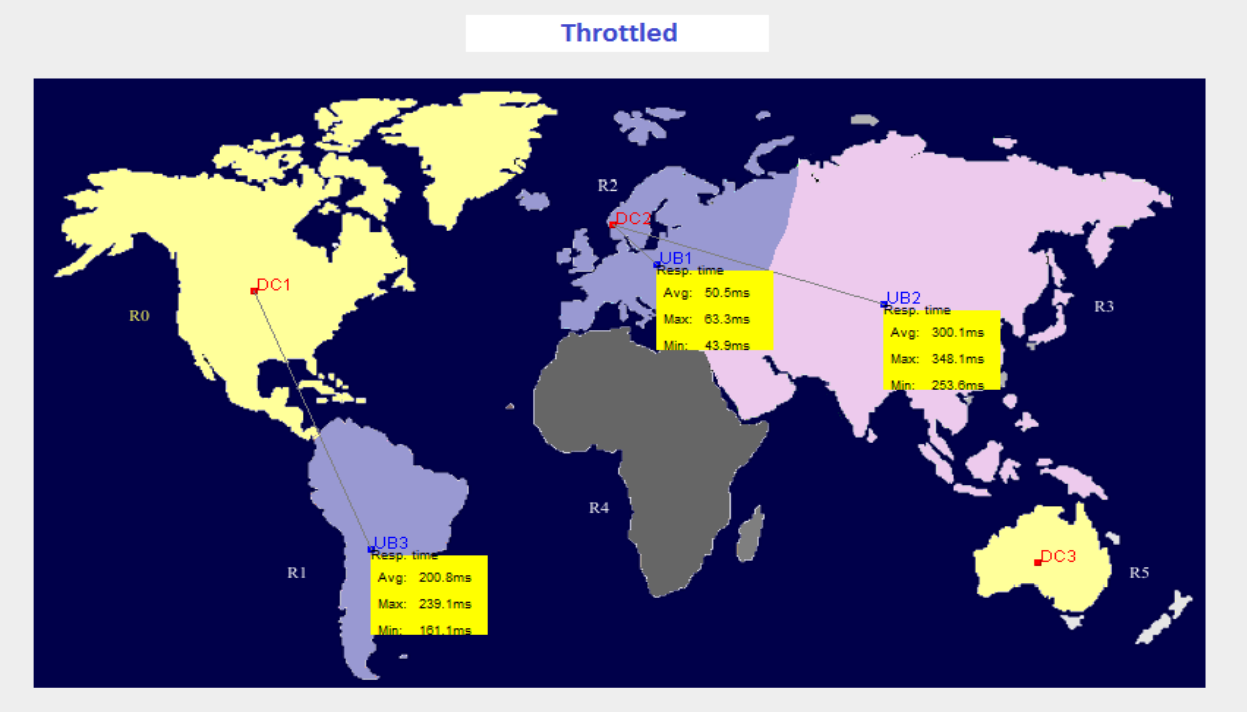


Cost

Total Virtual Machine Cost ($):	6.50
Total Data Transfer Cost ($):	0.19
Grand Total($) :	6.69

Data Center	**VM Cost $**	**Data Transfer Cost $**	**Total $**
DC1	3.35	0.13	3.48
DC2	2.75	0.06	2.81
DC3	0.40	0.00	0.40

5.2.3Throttled



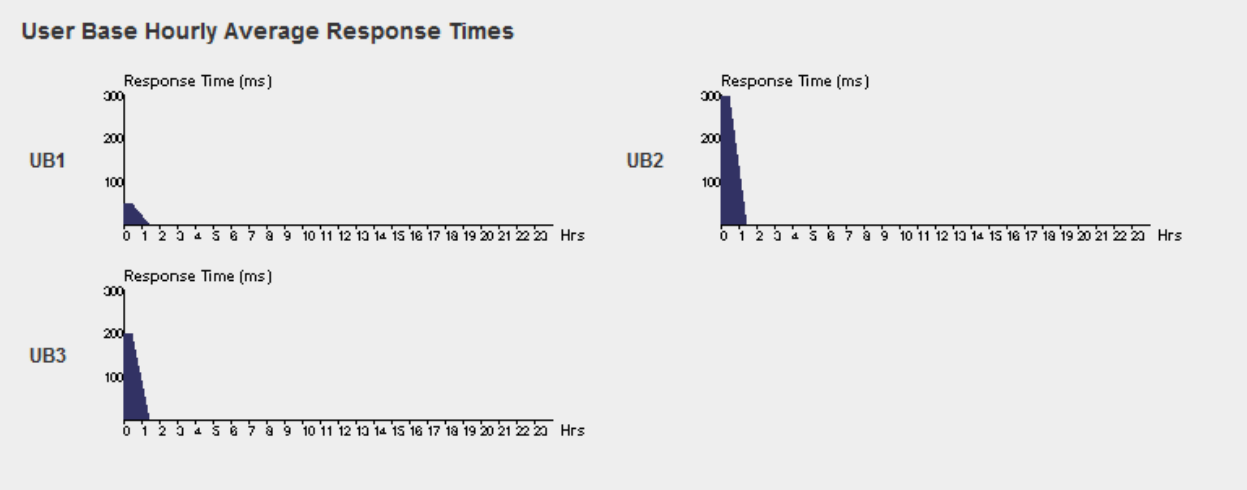
Overall Response Time Summary

	Avg (ms)	**Min (ms)**	**Max (ms)**
Overall response time:	181.70	43.88	348.14
Data Center processing time:	0.38	0.00	1.05

Response Time by Region

User base	**Avg (ms)**	**Min (ms)**	**Max (ms)**
UB1	50.49	43.88	63.31
UB2	300.11	253.63	348.14
UB3	200.80	161.11	239.12

User Base Hourly Response Times



Data Center Request Servicing Times

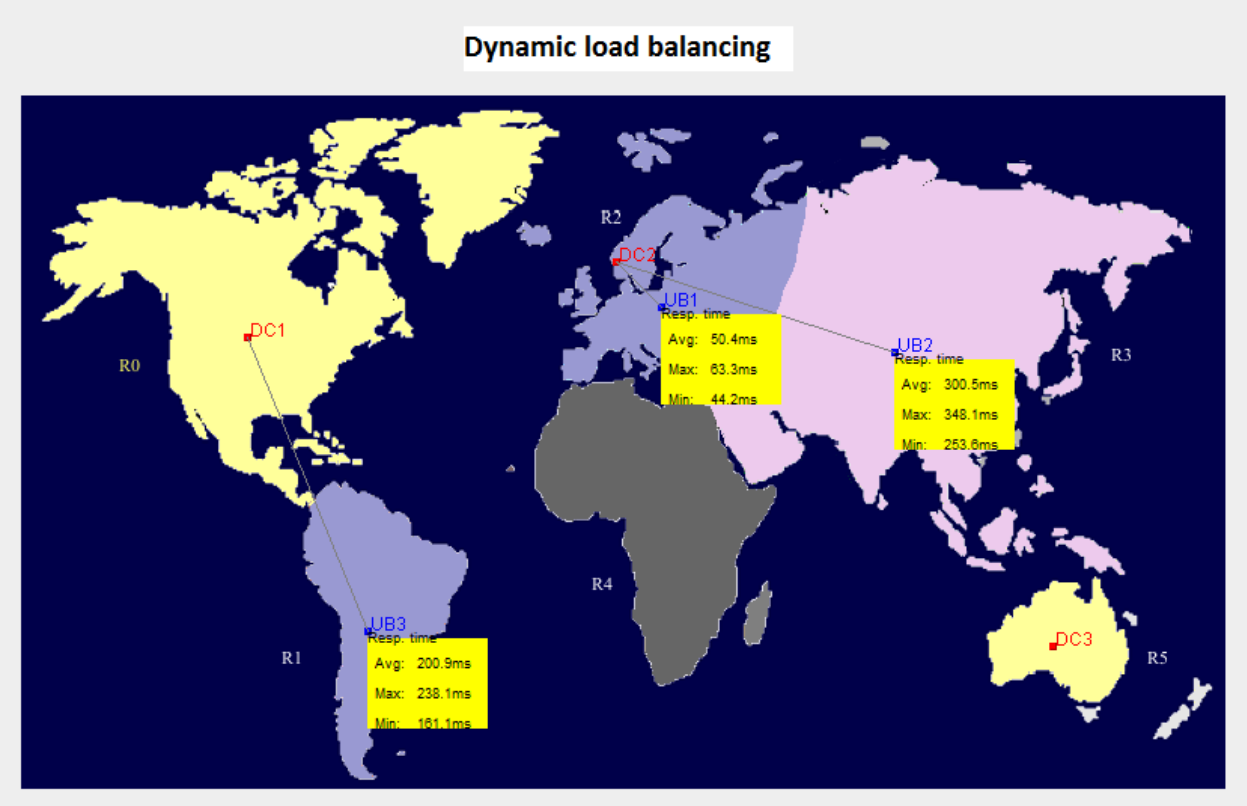
Data Center	**Avg (ms)**	**Min (ms)**	**Max (ms)**
DC1	0.10	0.02	0.11
DC2	0.51	0.02	1.05
DC3	0.00	0.00	0.00

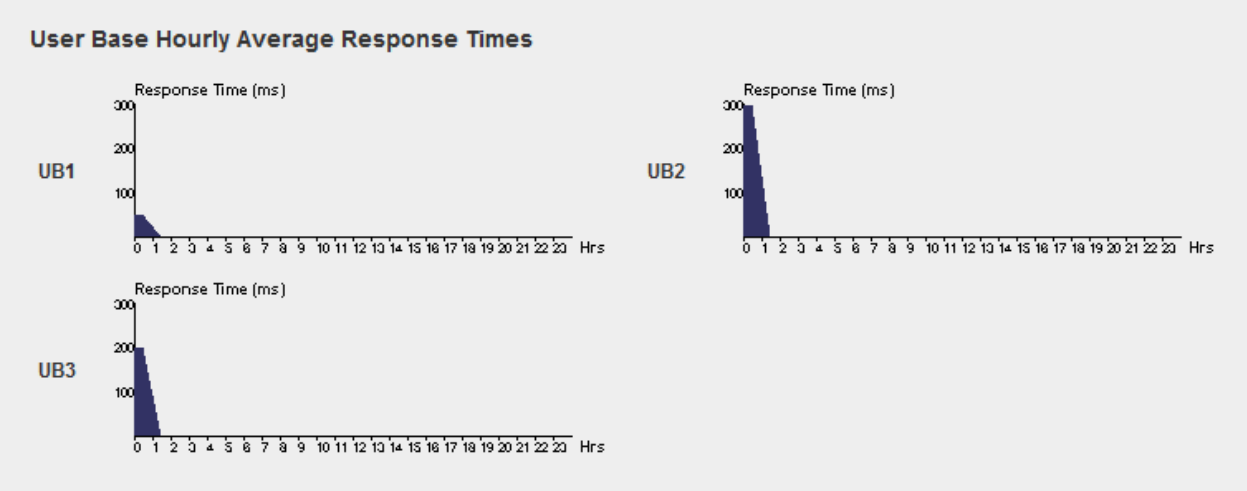
Cost

Total Virtual Machine Cost ($):	0.30
Total Data Transfer Cost ($):	0.03
Grand Total($) :	0.34

Data Center	**VM Cost $**	**Data Transfer Cost $**	**Total $**
DC1	0.14	0.02	0.17
DC2	0.09	0.01	0.10
DC3	0.07	0.00	0.07

5.2.4. Dynamic load balancing (Proposed method)





Overall Response Time Summary

	Avg (ms)	**Min (ms)**	**Max (ms)**
Overall response time:	181.67	44.23	348.14
Data Center processing time:	0.39	0.00	1.05

Response Time by Region

User base	**Avg (ms)**	**Min (ms)**	**Max (ms)**
UB1	50.50	44.23	63.31
UB2	299.89	253.63	348.14
UB3	200.94	161.11	241.12

Data Center Request Servicing Times

Data Center	**Avg (ms)**	**Min (ms)**	**Max (ms)**
DC1	0.10	0.02	0.11
DC2	0.51	0.02	1.05
DC3	0.00	0.00	0.00

Cost

Total Virtual Machine Cost ($):	0.30
Total Data Transfer Cost ($):	0.03
Grand Total($) :	0.34

Data Center	**VM Cost $**	**Data Transfer Cost $**	**Total $**
DC1	0.14	0.02	0.17
DC2	0.09	0.01	0.10
DC3	0.07	0.00	0.07

The following table shows the comparison of overall response time of the above listed four algorithms:

			Avg (ms)	Min (ms)	Max (ms)
Overall response time:		**Dynamic Load balancing**	181.67	44.23	348.14
Data Center processing time:			0.39	0	1.05
		Throttled	181.7	43.88	348.14
			0.38	0	1.05
		ESCE	184.82	40.63	365.12
			0.88	0	3.45
		Round Robin	181.73	44.23	348.14
			0.4	0	1.23

Table 1: the comparison of overall response time of four algorithms

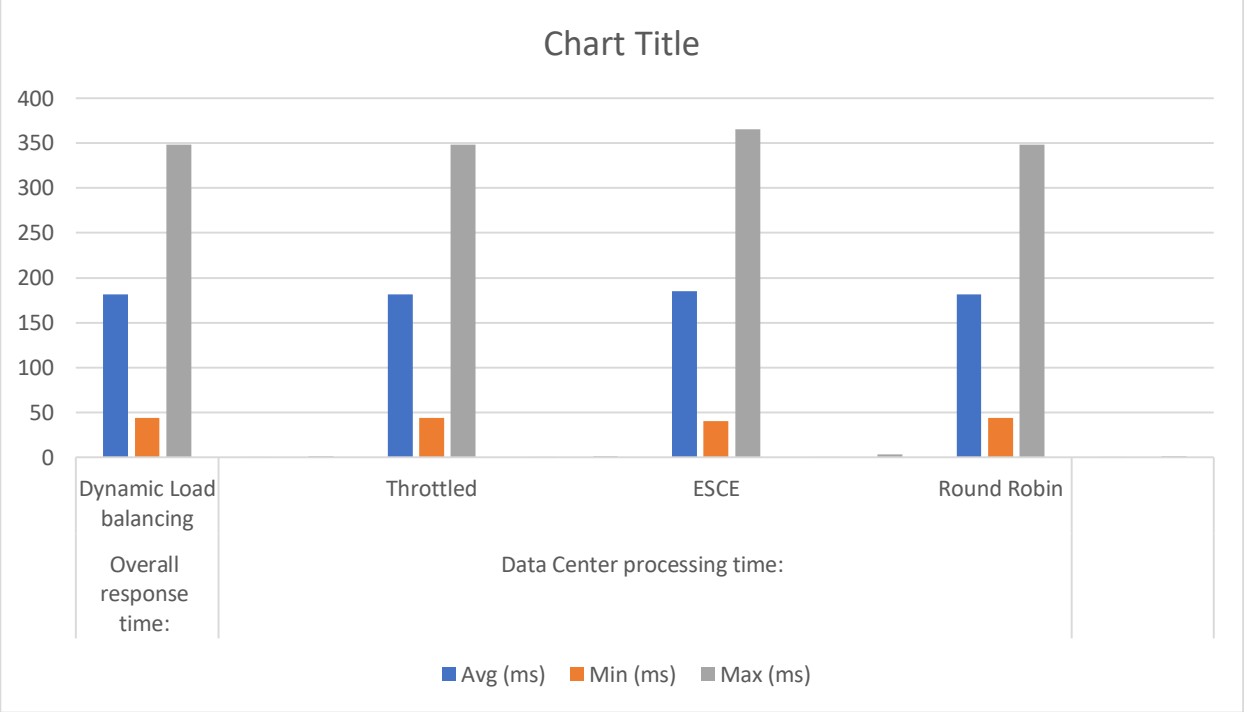


Figure 5.1: the comparison of overall response time of four algorithms

So, the above simulation results show and prove that our proposed dynamic load balancing method performs better than existing three methods.

Chapter VI

CONCLUSION AND FUTURE WORK

Chapter 6

CONCLUSION AND FUTURE WORK

6.1. Conclusion

In the cloud computing technology the topical trends explored this thesis. Load balancing and the live migration is remained the main issue. As a result, the load balancing should be more dynamic and competent to get better the performance of the cloud computing technology. In the load balancing method, we have to deal with the situation of dynamic loading of the workload. The existing work regarded as several load balancing procedures that handle the load distribution among different virtual machines and allocates load corresponding to their priority and states. There is an issue of overloading which means the resources may be over utilized that's why there increases the response time. There is an also concern of under loading means the resources may under-utilized and for this reason there may increase in the power consumption. According to the load balancing algorithm pedestal on resource utilization, the workload is distributed to the virtual machine stand on different parameters as well as speed, memory and power consumption of the virtual machines. The scrutiny of the results explains that response time of the algorithm is lessen as compared to the other algorithms. As a result the intended work is also energy efficient.

6.2. Future work

In future, the load balancing can be more dynamic by allocating the weights to the constraints dynamically in order to compute the priority of the virtual machine. In this thesis, response time of the algorithm is progress but the result of allocating different weight on response time can also be estimated and the result can be balanced in future. So thus, load balancing can be more dynamic.

REFERENCES

[1] 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.*

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

[3]

[4] 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.

[6] 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.

[7]

[8] 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.

[9] 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.

[10] B. M. Purcell, "Big data using cloud computing", in Journal of Technology

Research, pp. 1-8.

[11] 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.

[12] 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.

[13] A. T. Velte, T. J. Velte and R. Elsenpeter, "Cloud Computing: A Practical

Approach", in *Tata Mcgraw-Hill*, 2010 edition, pp. 8-11.

[14] T. Mather, S. Kumaraswamy and S. Latif, "Cloud Security and Privacy: An

Enterprise Perspective on Risks and Compliance", *O'Reilly Media*, 2009.
[15] 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.
[16] 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.
[17] 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.
[18] 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.
[19] 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.
[20] 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.
[21] 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.
[22] 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.
[23] 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.
[24] 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.
[25] 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.

[26] 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.

[27] 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.

[28] 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.

[29] 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.

[30] 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.

[31] 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.

[32] 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.

[33] 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.

[34] 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.

[35] 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.

[36] 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.

[37] 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.

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

[39] 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.

[40] 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.

[41] 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.

[42] 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.

[43] 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.

[44] 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.

[45] 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.

[46] 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.

[47] 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.

[48] 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.

[49] 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.*

[50] 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.