

**ADAMA SCIENCE AND TECHNOLOGY
UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING
AND COMPUTING
Department of Computing**



A MASTER'S THESIS

ON

**An Enhanced Active Monitoring Based Load
Balancing Algorithm in Cloud Environment**

MASTER OF SCIENCE IN SOFTWARE ENGINEERING

BY:

TEGEGN DITA

GSR/02987/07

ADVISOR: Dr. MESFIN ABEBE

May, 2017

ASTU

**ADAMA SCIENCE AND TECHNOLOGY
UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING
AND COMPUTING**

**An Enhanced Active Monitoring Based Load
Balancing Algorithm in Cloud Environment**

**A Thesis Submitted to School of Graduate Studies in
Partial Fulfillment of the Requirement for the Degree
of Master of Science in Computing**

By:

Tegegn Dita

Major Advisor: Dr. Mesfin Abebe (Asst. Professor)

May, 2017

ASTU, Ethiopia

**ADAMA SCIENCE AND TECHNOLOGY
UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING
AND COMPUTING**

**An Enhanced Active Monitoring Based Load
Balancing Algorithm in Cloud Environment**

**A Thesis Submitted to School of Graduate Studies in
Partial Fulfillment of the Requirement for the Degree
of Master of Science in Computing**

By

Tegegn Dita

Signature of the Board of Examiners for Approval:

<u>Name</u>	<u>Role</u>	<u>Signature</u>	<u>Date</u>
1. Dr. Mesfin Abebe (Asst. Prof)	Advisor	_____	_____
2. Dr. Tibebe Beshah (Asst. Prof)	Examiner	_____	_____
3. Dr. N.Satheesh Kumar (Asso. Prof)	Examiner	_____	_____
4. Mr. Dereje Regassa	Chair Person	_____	_____

Declaration

This research paper is my original work and has not been presented as a partial degree requirement for a degree of Master of Science in this or any other university. And that all sources of materials used for the thesis have been duly acknowledged and recent documents.

Tegegn Dita

Signature: _____

Date: _____

This thesis has been submitted for examination with my approval as university advisor.

Dr. Mesfin Abebe (Asst. Professor)

Signature: _____

Date: _____

May, 2017

ASTU, Ethiopia

Acknowledgement

First of all, I would like to forward my gratefully and deepest thanks to the almighty God for his guidance, help, support and who added days to my age, Glory to God. Next, I would like to express my profound sense of gratitude towards my advisor, Dr. Mesfin Abebe, Asst. Professor, Department of Computing and Engineering. For his able guidance, support and encouragement throughout the period this work was carried out. I am also greatly obligated to forward my gratitude to Dr. Ravindra Babu. Finally, I would like to thank everyone at the College of Electrical Engineering and Computing. Who encouraged and assisted me throughout the completion of this thesis.

Acronyms and Abbreviations

AMLBA	Active Monitoring Load Balancing Algorithm
API	Application Program Interface
Avg	Average
CPU	Central Processing Unit
DCC	Data Center Controller
ESCE	Equally Spread Current Execution
Exp	Experiment
FCFS	First Come First Serve
GB	Giga Byte
GUI	Graphical User Interface
IaaS	Infrastructure as a Service
ID	Identification
IDE	Integrated Development Environment
IT	Information Technology
MAMLBA	Modified Active Monitoring Load Balancing Algorithm
Max	Maximum
Min	Minimum
NIST	National Institute Standard Technology
RF	Reliability Factor
RRLBA	Round Robin Load Balancing Algorithm
SaaS	Software as a Service
SSL	Secure Socket Layer
TLB	Throttled Load Balancing
TVMLB	Throttled Virtual Machine Load Balancing
UB	User Base
VM	Virtual Machine
WAN	Wide Area Network

Table of Contents

Declaration.....	i
Acknowledgement	ii
Acronyms and Abbreviations	iii
List of Tables	vi
List of Figures	vii
Abstract.....	viii
CHAPTER ONE	1
1. INTRODUCTION	1
1.1 Background.....	1
1.2 Statement of the Problem.....	2
1.3 Research Question	3
1.4 Purpose of the Study	3
1.5 Objective of the Study.....	3
1.5.1 General Objective	3
1.5.2 Specific Objectives	4
1.6 Scope of the Study	4
1.7 Limitation of the Study	4
1.8 Significance of the Study	4
1.9 Organization of the Thesis	5
CHAPTER TWO	6
2. LITERATURE REVIEW	6
2.1 Cloud Computing.....	6
2.1.1 Features of Cloud Computing	9
2.2 Characteristics of Cloud Computing.....	10
2.3 Cloud computing and Virtualization.....	11
2.4 Load Balancing Algorithm in Cloud Computing.....	12
2.4.1 Round Robin Scheduling Algorithm.....	14
2.4.2 Active Monitoring Load Balancing Algorithm.....	17
2.4.3 Throttled Load Balancing Algorithm.....	20
2.5 Optimization of Resources in Cloud Computing Using Effective LBAs	24
2.6 A Comparative Study of Load Balancing Algorithms in Cloud Computing	26

2.7	Efficient Load Balancing Algorithm in Cloud Environment	26
2.7.1	Load Algorithm Active VM Load Balancing [31].....	27
2.8	Related Work	30
CHAPTER THREE		34
3.	METHODOLOGY OF THE RESEARCH	34
3.1	Research Design.....	34
3.2	Data Source.....	36
3.3	Data Collection Techniques	39
3.4	Issues of Reliability and Validity	40
3.5	Sampling Techniques	41
3.6	Open Source Cloud Computing Tools	42
3.6.1	CloudAnalyst	42
3.6.2	Cloudsim	43
3.7	Evaluation Approach	44
CHAPTER FOUR.....		45
4.	DESIGNING THE MODIFIED ACTIVE MONITORING LOAD BALANCING ALGORITHM..	45
4.1	Proposed Using Load Balancing Algorithm VM-Assign	45
CHAPTER FIVE		50
5.	PRESENTATION, ANALYSIS AND INTERPRETATION OF DATA	50
5.1	Implementation Plan for the Proposed Algorithm	51
5.2	Simulation Setup	56
5.3	Outputs and Results	56
5.4	Discussion and Interpretation.....	68
CHAPTER SIX.....		70
6.	CONCLUSION AND RECOMMENDATION	70
6.1	Conclusion	70
6.2	Recommendations	72
6.3	Future Work	72
References.....		73
Appendixes		x

List of Tables

Table 1: Source: (John Ribeiro , May 27,2014).....	37
Table 2: Parametres and values in CloudAnalyst simulator	56
Table 3: Overall response time for case-1	57
Table 4: Overall data center processing time for case-2	59
Table 5: Overall response time for case-3	61
Table 6: Overall data center processing time for case-4	62
Table 7: Overall response time for case-6	63
Table 8: Overall data center processing time for case 7	64
Table 9: Overall response time for case 8.....	65
Table 10: Overall data center processing time for case 9	66

List of Figures

Figure 1: Basic Cloud Environment [3]	7
Figure 2: Cloud Computing Architecture [14].....	8
Figure 3: Three components make up of a cloud computing solution [3]	11
Figure 4: Cloud Load Balancer [23]	14
Figure 5: Round robin algorithm [3].....	16
Figure 6: Active monitoring algorithm [44]	19
Figure 7: Throttled load balancing algorithm [4].....	22
Figure 8: Equally spread active execution load to the cloud system	27
Figure 9: Twitter users in 2014. Source: (John Ribeiro, May 27, 2014)	41
Figure 10: Architecture of CloudAnalyst simulator [27].....	42
Figure 11: VM Load Balancer Using Cloudsim Library	43
Figure 12: Flow of existing active monitoring load balancing algorithm.....	46
Figure 13: Flow of the Proposed VM-assign Algorithm	47
Figure 14: Output Screen after Run Simulation.....	54
Figure 15: NetBeans IDE for Request Assignment	54
Figure 16: Simulation Result for case one	58
Figure 17: Simulation Result for 3 UBs and 6 DC	58
Figure 18: Simulation Result for 30 UBs.....	60
Figure 19: Overall response time summary for 30 UBs	60
Figure 20: Final test result	67
Figure 21: Final test output in percent	68

Abstract

In current world scenarios the cloud computing is becoming popular these days due to its high availability and applicability. Because of the Internet daily increasing traffic introduces the requirement of the load balancing concept to get the most utilization of the resources available on cloud. A number of algorithms have been proposed to assign the client's request to the available cloud nodes and to provide efficient mechanisms. And the aim is to increase the overall performance of the cloud and provide more satisfaction to users and efficient services.

Initially this study gives an introduction to cloud computing and load balancing algorithms. A detailed research analysis is made on the modification of active monitoring load balancing algorithm and the utilization concept of an active monitoring load balancing algorithm. To do so a number of methods are applied: like that of using a java language to modify the load balancing algorithm and then by adding the proposed algorithm to the CloudAnalyst tool in order to fulfill the study. To analyzed and develop the concept of active monitoring load balancing algorithm. Now days there are number of recent literatures on cloud computing in general and its deployment mechanisms in particular were well studied.

The result of the study shows that, considering an active monitoring load balancing algorithm by proposed an improved active monitoring load balancing algorithm in cloud computing environment. At the time of increasing load on a virtual machine the complexity of the load balancing algorithm increases. So that to increase its performance and to reduce the complexity of the load balancing algorithm we used buffer. Based on this we reduced the complexity of active monitoring load balancing algorithm at the time of number of request increases on a virtual machine. To apply all the above mentioned concepts and to increase its performance depend on the service broker policy based on the overall response time and overall data center processing time.

Keyword: Active monitoring, CloudAnalyst, Cloud computing, Cloudsim, Load balancing, Virtual machine balancing.

CHAPTER ONE

1. INTRODUCTION

1.1 Background

Daily increasing traffic on internet is the case for the requirements of the load balancing concept to effectively utilized resources on the cloud. Cloud is subject and has direct effect on user consumption of resources controlled by cloud provider to the user requirement, load balance and other constraints [1, 2]. To utilize completely the cloud computing we need an effective and efficient load balancing algorithm in cloud computing environment.

In cloud computing environment balancing load is one of the important issues. Now days, cloud computing is one of the most popular and rapidly growing technology. From all over the world clients request for various cloud computing services. In present situation the load balancing algorithms are built to allocate the user requests/tasks to cloud resources in efficient and intelligent way [3].

Every field of our life serves with the cloud computing environment. But while fulfilling number of our requests it faces few limitations to should be overcome. Along with it asks us to focus on issues like security, authentication, fault tolerance, load balancing, and availability [3]. One of the cloud computing service providers' objective is the establishment of an effective load balancing algorithms in cloud computing environment. And the other is how to use cloud computing resources efficiently. While balancing the load, certain types of information such as the number of jobs waiting in queue, job arrival rate, CPU processing rate at each processor, as well as at neighboring processors and the like may be exchanged among the processors for improving the overall performance. Good load balance will improve the performance of [4] the entire cloud in cloud computing environment.

There is no common mechanism that can adapt to all possible different situations and various methods have been developed to improve the existing solutions to resolve new problems [3].

However each particular method has advantage in a particular area but not in all situations. Therefore, the current model integrates several methods between the load balance methods based on the system status.

The existing active monitoring load balancing algorithm has been modified to achieve better overall response time and data center processing time [4].

In this study the algorithm proposed to distributes efficiently the requests/tasks to all the VMs using the active monitoring load balancing algorithm for their execution, considering the CPU utilization of all the virtual machines.

1.2 Statement of the Problem

Today using the load balancing concept number of websites work, in this case our sites running on several machines with several instances. Users don't send requests/tasks to these machines directly; instead they send requests/tasks to another machine (load balancer). In cloud computing, load balancing is a mechanism to distribute workload on the multiple machines or a computer cluster through network links to achieve optimal resource utilization for maximizing throughput and minimizing overall response time [3]. And also used for avoiding too much overload on the resources. In cloud computing, it is the scheduling of different tasks of jobs which are expected to be executed over different datacenters. This distribution should give assurance to the minimum execution time of the overall tasks.

The active monitoring algorithm load balances the requests between that of the available virtual machines. In a way that manages almost same number of active requests/tasks on each virtual machine at any given period of time. On the other hand the throttled algorithm checks for suitable and available virtual machine in the list of VMs along with the state (busy/ available). The active monitoring load balancing algorithm is similar to the throttled load balancing algorithm [3]. But, in active monitoring load balancing algorithm as it allocates or assigns multiple tasks/requests on a VM the complexity increases.

In earlier load balancing algorithm on in existing methods only how to balance load is stated, but in those schemas only some factors are considered like allocation of requests/tasks on a virtual machine with least loaded virtual machine and the like.

But other factors like if, the number of requests/tasks allocation increase on a virtual machine, if all virtual machine reached in maximum limitation to accept any requests/tasks, the overload of the virtual machine or the complexities of the virtual machines and the utilization concepts of the algorithms are not considered [21, 35]. This factor is the case in order to decrease the performance of the active monitoring load balancing algorithm. Because the number of requests/tasks to allocate on a virtual machine.

This study tries to analyze the benefits in terms of resource use, minimize overall response time, datacenter processing time and avoid overload in cloud computing environment.

Further the main focus of this study was to found the most modified active monitoring load balancing algorithm. To do so by reducing the complexity of the active monitoring load balancing algorithm at the time of number of requests/tasks allocates on a virtual machines. And to give the utilization concepts for active monitoring load balancing algorithm at virtual machine level in cloud environment.

1.3 Research Question

- How to solve the complexity of an active monitoring load balancing algorithm in cloud computing environment?

1.4 Purpose of the Study

The purpose of the study was to provide a solution for the complexity of active monitoring load balancing algorithm in order to distribute the workload efficiently on the available virtual machines. And ensuring that over utilization of the resources/virtual machines will not occur in the cloud system, and also to give the utilization concept of active monitoring load balancing algorithm.

1.5 Objective of the Study

1.5.1 General Objective

The general objective of this study was, to modify the active monitoring load balancing algorithm to improve its performance whenever multiple requests/tasks allocate on a virtual machines, and to give the utilization concept of active monitoring load balancing algorithm in cloud computing environment.

1.5.2 Specific Objectives

- To study the existing VM load balancing algorithms.
- To understand and describe about the active monitoring load balancing algorithm.
- To increase the performance of active monitoring load balancing algorithm.
- To modify a reliable, scalable and flexible monitoring solutions in load balancing algorithm on a virtual machine.
- To give comparison of the proposed algorithm with the existing algorithms.

1.6 Scope of the Study

The main intent of the study is to consider the concept of active monitoring load balancing algorithm in cloud computing environment. To consider and prove solution the complexity of active monitoring load balancing algorithm at the time of requests/tasks allocates on a virtual machines. This study doesn't consider about the other types of load balancing algorithms at virtual machine level in cloud environment.

1.7 Limitation of the Study

There are limitations in proposing active monitoring load balancing algorithm in cloud computing environment. These limitations are raised because of lack of experience of research, lack of materials to support the study, shortage of on regular training and others. But these types of problems raised we can minimize the challenge.

1.8 Significance of the Study

This research study could allow giving concepts for users about active monitoring load balancing algorithm with better processing time, response time, cost effectiveness and other service delivery. And also it gives novel vision to concerned customer towards better and modernized concept in cloud computing environment.

On the same way the proposed active monitoring load balancing algorithm can be used to reduce the complexity of it whenever number of requests/tasks loads on virtual machines.

At this time by using buffer (used to temporarily stored data) which used to put the request on it until some of the virtual machines reduce the load to allocate on it.

Furthermore the study could be used as a reference point for further research and studies for new active monitoring load balancing algorithm strategies and frameworks to be considered all over the world.

1.9 Organization of the Thesis

The research consists of six main chapters; the rest of the research is organized as follows: Chapter Two, presents the conceptual discussion and detailed literature review about load balancing algorithms in cloud computing environment. In this section discusses about the technical issues which are related in cloud computing environment and the most known contributions for the load balancing algorithms. The chapter also well introduced the different types of load balancing algorithms for improved quality of service in the cloud. Finally this chapter includes some related and recent research works.

Chapter Three, this chapter describes about the methodology of the research and strategy to identify the load balancing algorithms solutions. That could be provided to cloud computing environment. In this section describes about research design, backgrounds of the algorithm for building the proposed load balancing algorithm for cloud computing. The architectural components of the implementing algorithm, sampling techniques, process model, data sources, open source cloud computing tools and issues of reliability and validity.

Chapter Four, presents about the designing of the modified active monitoring load balancing algorithm at virtual machine level in cloud computing environment.

Chapter Five, describes about the presentation, analysis and interpretation of data. The implementation plan for the proposed algorithm, the simulation setup and the results of the outputs are also discussed.

Finally, chapter six includes about, conclusions and recommendations of future works are presented in the study.

CHAPTER TWO

2. LITERATURE REVIEW

In this section to understand the technical issues which are related in cloud computing environment. And to discuss about the most known contributions for the load balancing algorithms as well as to briefly summarize the load balancing algorithms used in the cloud computing environment.

2.1 Cloud Computing

“Recently, the cloud computing become one of a new global trend of computing. It is a modern style of using the power of internet and WAN to access resources remotely. To achieve high availability, flexibility, cost reduced and on demand scalability, it is a new solution and strategy” [5, 6].

Cloud computing has become very popular in recent years as it offers greater flexibility and availability of computing resources. Cloud computing is a general term used to describe a new class of network based computing that takes place over the internet. And also it is a new paradigm in which both new technologies and existing ones converge to offer as services all the capabilities of a computing system to different customers.

These services can be accessed from any device with internet connection independently of their physical location. As [7] explained about cloud computing, *“it is widely spread technique which has emerged a new technology. And that provides large amounts of computing and data storage capacity to its users with a promise of increased scalability, better availability, and reduced administration and maintenance costs”*.

According to [8] explanation about cloud computing, it is also intended to enable the computing across largely and diverse resources and dynamic stability. It described a large number of applications that are accessible through the internet and for this purpose large data servers to host web applications are used. It provides an on-demand computing that highlights subscription based services where one can obtain huge number of computer resources and the storage spaces. Cloud computing provides scalable, pay-as-we-go, just in time allocation of resources.

Cloud computing provides a new deployment model of resources as accessible using public or private networks [3].

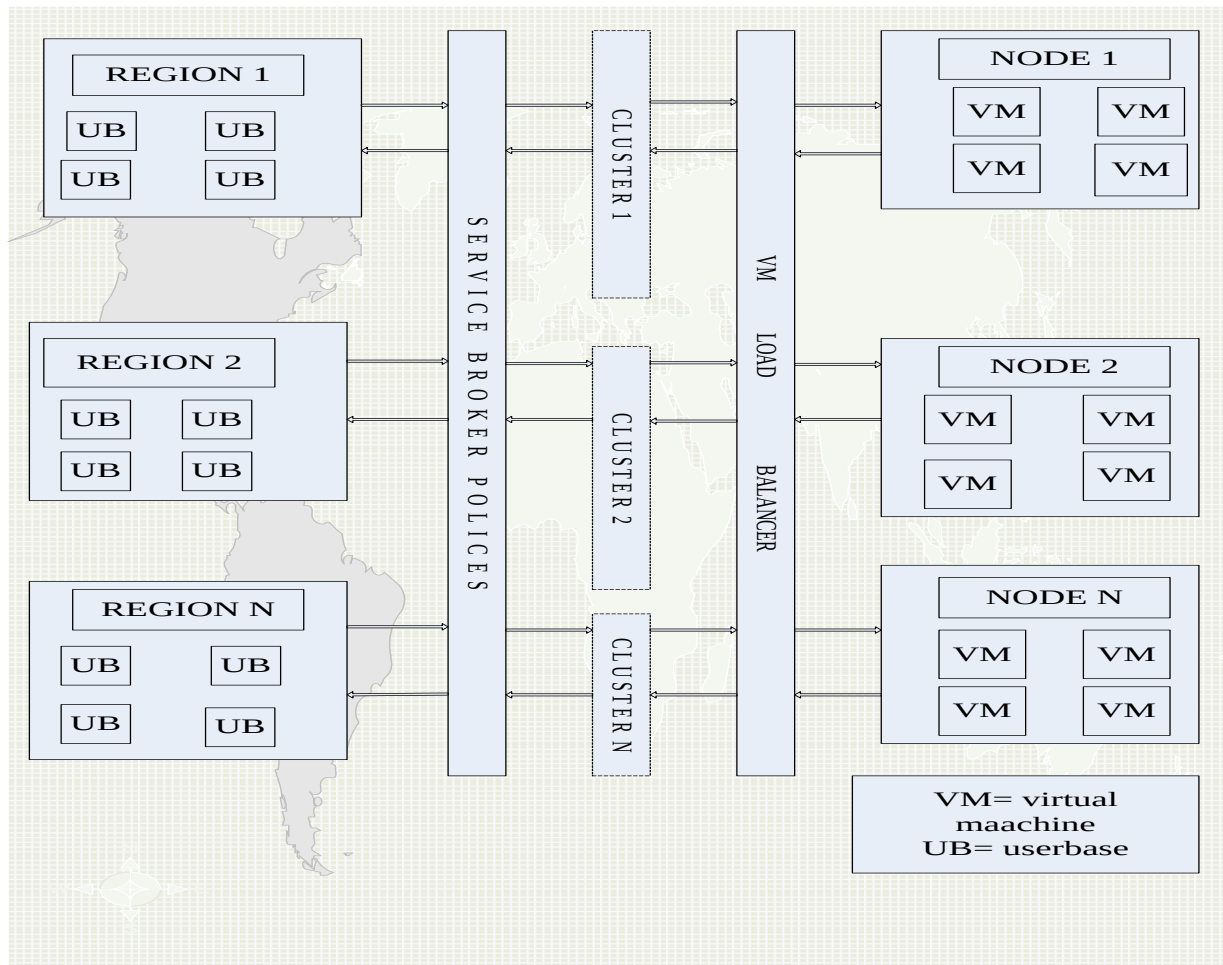


Figure 1: Basic Cloud Environment [3]

According to the client's requirement at specific time the cloud computing is an on demand service in which infrastructure, platform and software are provided on demand. It is a term generally used in the case of internet. One can view whole internet as a cloud. Thus the all the above mentioned services are access by a user as a client to the cloud. Now as the basic idea of cloud computing is to provide resources such as virtual machines as services on demand [9, 10].

The cloud computing services are using pay-as-use model to users, regardless of their location. And it is a cost effective model for provisioning services and it makes IT management easier and more responsive to the changing needs of the business [11].

Cloud computing is internet based computing, whereby shared resources, software and information are provided to computers and other devices on-demand, like a public utility [12].

And also it is a model for delivering information technology services in which resources are retrieved from the internet through web-based tools and applications, rather than a direct connection to a server. Data and software packages are stored in servers. However, cloud computing structure allows access to information as long as an electronic device has access to the web. This type of system allows employees to work remotely [13]. “Cloud computing consist of four different layers, each layer having its own functionalities”. Moreover the services provided by the cloud computing are also mentioned in the below figure.

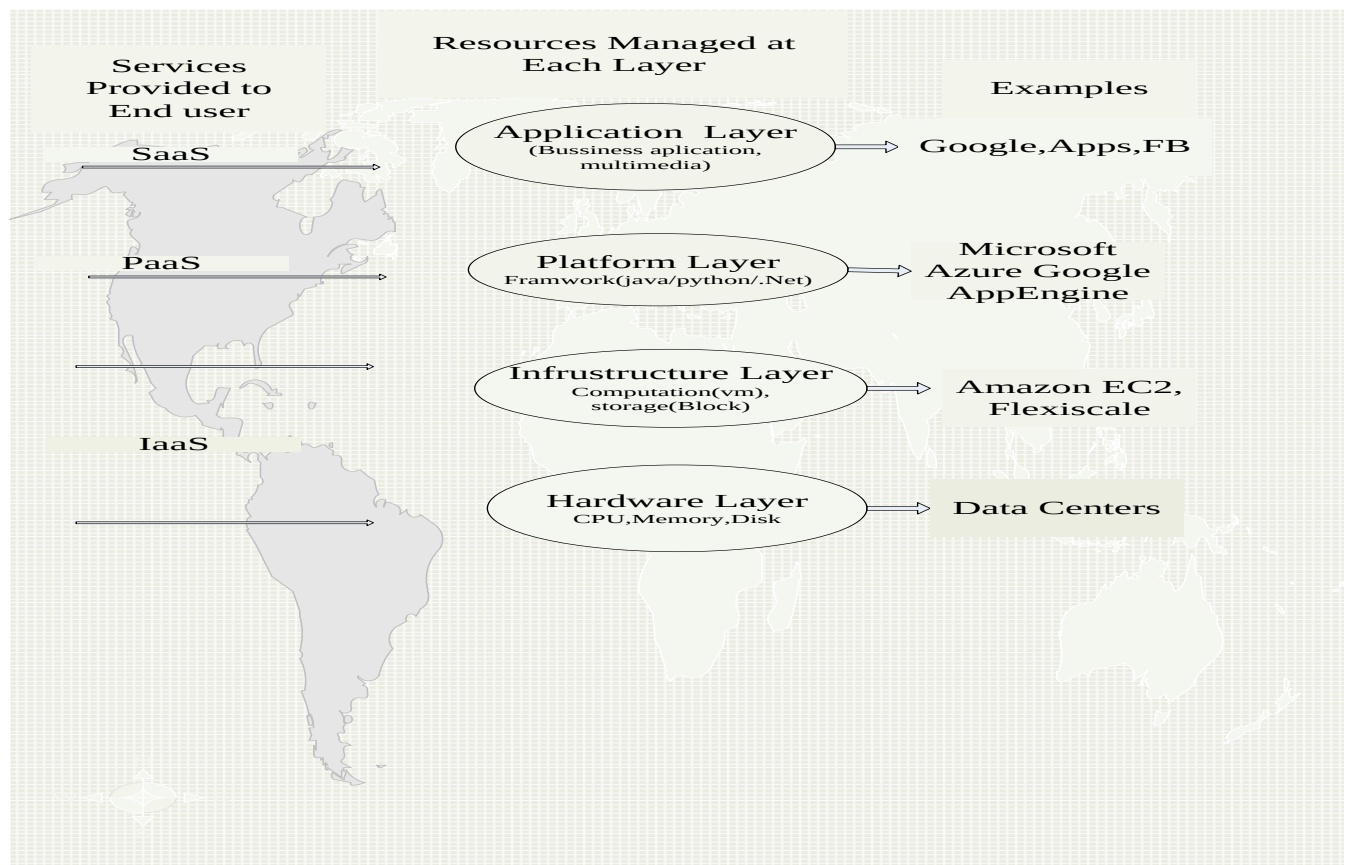


Figure 2: Cloud Computing Architecture [14]

Lot of work is being carried out to achieve load balancing in cloud infrastructure now days. There are so many load balancing algorithms performing their task on different layers of cloud with different level of complexities.

To achieve great level of load balancing, so many people use many complex algorithms. But as a result it adds more processing load on the system executing such logic [3].

2.1.1 Features of Cloud Computing

In cloud computing environment the features of cloud computing like that of (Resource pooling, Rapid Elasticity, Scalability, On-demand self-service, Efficiency and Dynamic Resource Allocation). In similar way described for the approach of load balancing which is static and dynamic.

A. Static Approach:

Design or implementation of the system is mainly defined in the static approach. The static load balancing algorithm divides the traffic equivalently between all users. It uses only information about the average behavior of the system. These are much simpler and ignore the current state or the load of the node in the system.

B. Dynamic Approach:

The current state of the system in dynamic approach is considered the load balancing decision. It is more suitable for widely distributed system such a cloud computing. Dynamic load balancing algorithms are capable of managing the dynamic change in load pattern in cloud environment [5, 6, 15].

Load balancing helps in proper utilization of resources and improve the performance of the system. Load balancing is a major issues in cloud computing. And also examined some existing load balancing algorithm that maintains load balancing and provide better scheduling and resource allocation techniques.

2.2 Characteristics of Cloud Computing

As the Sun Microsystems and (NIST) describes about the characteristics of cloud computing, *“cloud services exhibit the following essential characteristics that demonstrate their relation to and differences from traditional computing approaches”*:

On –demand

On-demand is one of the characteristics of cloud computing. With a service provider no need to require human interaction. In on-demand, instead of requiring a long-term contract for services with an IT organization or a service provider, clouds work on needed model where an application may exist to run a job for a few hours. On the other hand it may exist to provide services to customers on a long-term basis.

Resource Pooling

In cloud computing the provider’s computing resources (such as: storage, processing, network bandwidth, and virtual machines) are pooled in order to serve multiple users using a multi-tenant model. Thus, different physical and virtual resources dynamically assigned and reassigned according to user demand. Between different parts of the same organization private clouds tend to pool resources.

Rapid Elasticity

In cloud computing cloud allows the extraction of computing resources and dynamic integration. Cloud services are available for provisioning, often appear to be unlimited and can be purchased in any quantity at any time. In cloud computing elasticity is an essential core feature of cloud systems. And also it circumscribes the capability of the underlying infrastructure to adapt changing. This cloud capability allows service consumers to create an elastic environment that can expand and contract based on the workload and target performance parameters.

Measured service

Automatically, cloud system control and optimize resource usage by leveraging a metering capability at some level of abstraction appropriate to the type of service. Such as: (storage, processing, bandwidth, or active user accounts).

Resource usage can be reported, monitored, and controlled to providing transparency for both the provider and user of the service. Based on cloud computing metering and accounting, users pay only for service they used. So that measured service billing is based on resource consumptions.

Virtualization

It is an essential technology which hides the technological complexity from the user and enables enhanced flexibility. And some features of the virtualization as follows:

- **Ease of use:** It is easier for the user to develop new applications, as well as to reduce the overhead for controlling the system.
- **Infrastructure independency:** By making the code platform independent, virtualization principle allows for higher interoperability.
- **Flexibility and Adaptability:** The underlying infrastructure can change more flexible according to different conditions and requirements by exposing a virtual execution environment.
- **Location independence:** In cloud computing services can be accessed independent of the physical location of the user and the resource.

2.3 Cloud computing and Virtualization

In cloud computing system the 3 (three) major components such as clients, data center, and distributed servers has a definite purpose and plays a specific role [3].

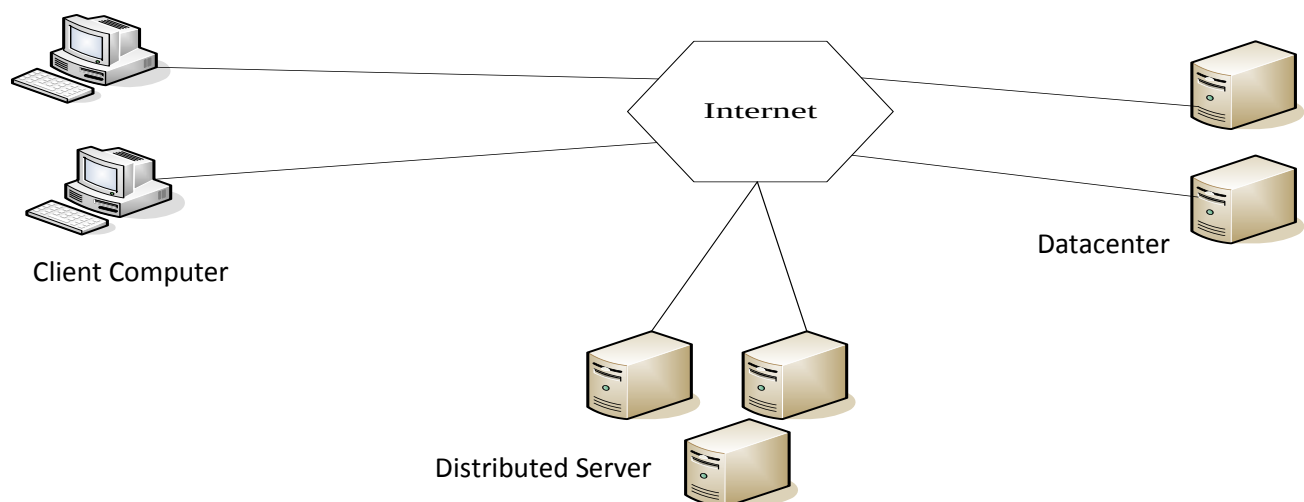


Figure 3: Three components make up of a cloud computing solution [3]

2.4 Load Balancing Algorithm in Cloud Computing

Load balancing is one of the main issues related to cloud computing. The load in this computing environment may be a memory, processor capacity and network. It is always required to share workload among the various nodes of the distributed system to improve the resource utilization and for better performance of the system [16, 17]. Load balancing in cloud computing systems is really a challenge now.

Always a distributed solution is required, because it is not always practically feasible or cost efficient to maintain one or more idle services just as to fulfill the required demands. Jobs can't be assigned to appropriate servers and clients individually for efficient load balancing as cloud is a very complex structure and components are present throughout wide spread area [4].

“Load balancing is a methodology that tends to distribute workload across a number of computers, or other means over the network links to achieve optimal resource utilization, minimum response time, to maximize throughput, and avoid overload” [18, 19, 20]. The load balancing algorithms are designed to balance the load in the system for the fulfillment of the goals and producing the optimized results as a whole.

According to [14], *“the load balancing is an efficient and critical concept in cloud computing and it helps in utilizing the resources optimally. Therefore it is used to minimizing the consumption of resources”*.

It is a mechanism that distributes the dynamic local work load. Evenly across all the nodes in the whole cloud to avoid a condition in which some of the nodes are over loaded, while some others are idle or under loaded. It helps to achieve a high user satisfaction and resource utility of the system. It also ensures that every computing resource is distributed efficiently and fairly with the load balancing [21]. The load can be balanced by active transferring of local workload from machine to the other machine in the remote node or machine that is used less.

Load balancing is also required in the clouds to meet the green computing. In this way, it can reduce the amount of energy by avoiding excessive interaction or virtual machines considering the workload [22].

Load balancers can be placed in the network to direct server requests according to server performance and the method of traffic distribution chosen. E.g. round robin. In certain cases network managers prefer to place the network load balancers outside the cluster to provide for increased horizontal scalability. And also it is one of the prerequisites to utilize the full resources of parallel and distributed systems.

On the other hand it is a technique that helped networks and resources by providing a maximum throughput with minimum response time. Load balancing is dividing the traffic between all servers, so data can be sent and received without any delay with load balancing in cloud computing environment [21].

In cloud environment many algorithms are available that helps in proper traffic loaded between all available servers. Load balancing is used for avoiding too much overload on the resources and dividing the traffic between servers and data [23]. And also used for minimizing the total waiting time of the resources. In cloud computing load balancing algorithms are used for balancing the load on virtual machine and cloud resources.

WORLD WIDE WEB

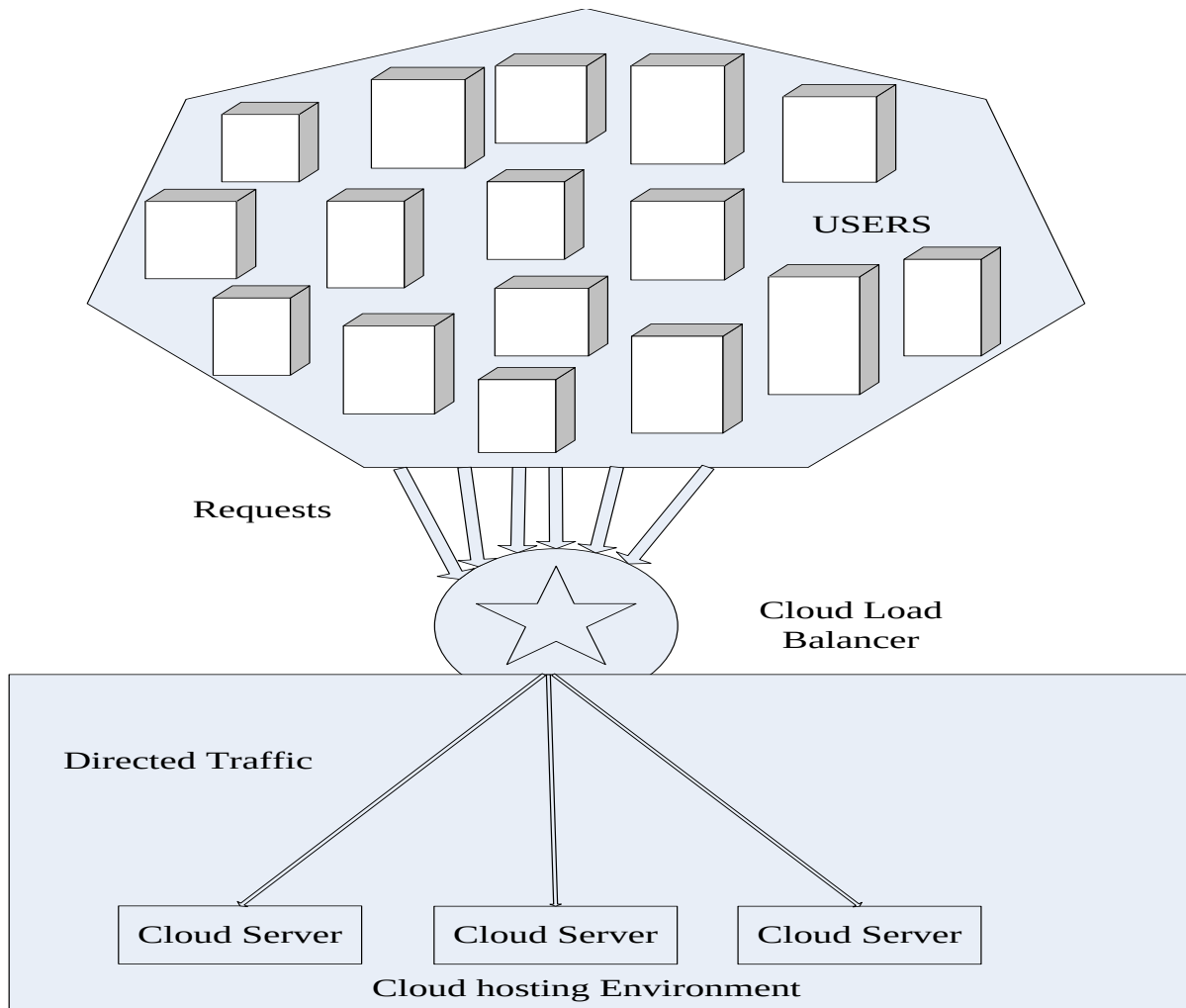


Figure 4: Cloud Load Balancer [23]

2.4.1 Round Robin Scheduling Algorithm

Among the load balancing the round robin load balancing algorithm is one of the simplest load balancing techniques, in which all processes are divided among all available processors. The allocation order of processes is maintained locally which is independent of the allocation from the remote processor. As the name indicates that these types of algorithm works in the round manner. Where each node is allotted with a time interval and has to wait for their turn and the time is divided and interval is allotted to each node. And then each node is allotted with a time interval in which they have to perform their tasks/requests.

When we compare this algorithm to the other algorithms the complexity of this algorithm is less. In this algorithm it simply allots the tasks/requests in round robin style which means doesn't consider the load on different machines [4]. In round robin load balancer [21], among the available virtual machines the data center controller picked randomly the virtual machine and then assigns the first request to that virtual machine. And also assigns the requests to the VMs in circular order. After the request assigned to the picked virtual machine, then the virtual machine is moved to the end of the list in the virtual machine list.

According to [47] out of all available algorithms the round robin load balancing algorithm is the simplest algorithm for load balancing. So that it never require the complex implementations of algorithm. Depending on the time scheduling manner for that of the incoming requests the algorithm simply maintains a queue and allocates them to the virtual machine.

In this algorithm each request allowed to be executed for that of specific time quantum. If the request is complete it allows other process of the virtual machine based on the algorithm. Here if it is still the execution incomplete, the algorithm has to wait for its next round.

According to [25] each node is given a quantum and operation. The resources of the service provider are provided to the clients request on the basis of time interval. So that we concluded that the round robin uses the time slicing mechanism.

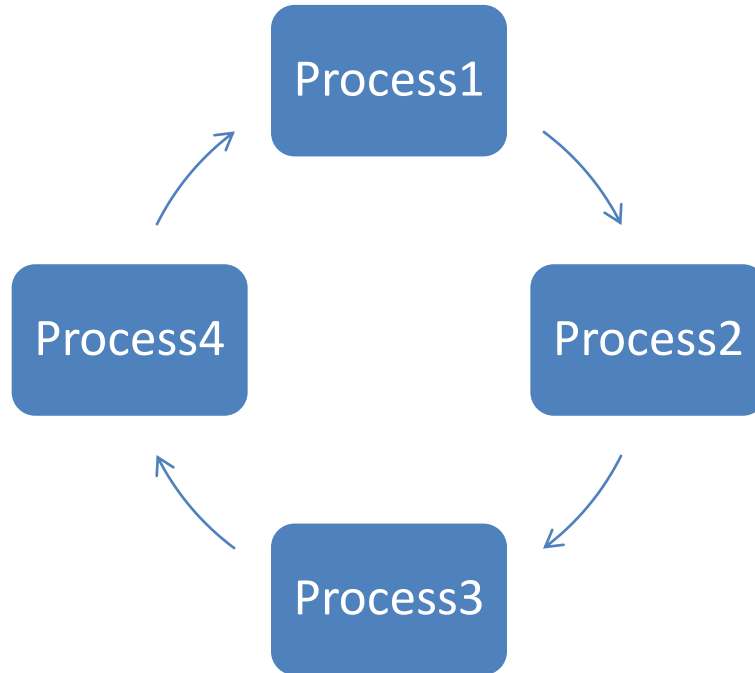


Figure 5: Round robin algorithm [3]

In the same way in the round robin load balancing algorithm as [46] explained about the techniques of this algorithm, the load balancing technique uses the concept of round robin. The method in this algorithm used the quantum. The round robin load balancing algorithm proceeds in the following way:

```
RRLBA ();  
{
```

From the virtual machine state list all available VM to be present. After this, without any values the hash maps starting. And then perform the while loop in the following way:

```
While (while loop condition: DCC receives the new request)  
Do {
```

Using the DCC requests are queued. After this the request removed from the beginning of the queue by the DCC and then, the decision making performed.

If (the hash map contains any of the entry of a VM which corresponds to the UB that is being currently requested and compare the VM status (whether the VM available or not)

{

Here the virtual machine is allocated to that particular user call;

}

else

{

Using the algorithm the VM is allocated to the UB request;

Entry of the UB and VM updated in the hash map as well as virtual machine state listing;

}

}

}.

2.4.2 Active Monitoring Load Balancing Algorithm

In cloud environment the random arrival of load can cause some of the server to be heavily loaded while other server is idle in equally spread current execution. Equally load distributing improves performance by transferring load from heavily loaded server to lightly loaded servers. Efficient scheduling and resource allocation is a critical characteristic of cloud computing based on which the performance of the system is estimated. The considered characteristics have an impact on cost optimization, which are obtained by improved response time and processing time. Here the jobs are submitted by the clients to the computing system. The cloud manager estimates the job size and checks for the availability of the virtual machine and also the capacity of the virtual machine. Once the job size and the available resource (virtual machine) size match, the job scheduler immediately allocates the identified resource to the job in queue.

The active monitoring load balancing algorithm is dynamic in nature. It keeps information about each virtual machine's and the number of request presently assigned to which virtual machine when a request is to allocate a new virtual machine reaches.

If there are more than one virtual machine, the first recognized is selected and the AMLB returns the virtual machine identity number to the datacenter controller. The datacenter controller sends the request to the virtual machine known by that VM identity number.

The datacenter controller warns the active monitoring load balancer to new allocation and the request is sent to it [24]. This type of load balancing algorithm works on the requests between available virtual machines. And also manages almost the same number of active requests on each virtual machine at any given time.

The [25] have developed the active monitoring load balancing algorithm which is used to maintain information about each VMs and the number of requests currently allocated to which VM.

The active monitoring load balancing algorithm stabilities the total available works in the total existing virtual machines. In such a manner that, these types of load balancing algorithm tries to balance total existing active works. Those are running on any virtual machine at any given time.

The active monitoring spread spectrum technique in which the load balancer spread the load of the work in hand into multiple virtual machines. The load balancer maintains a queue of the works that need to use and are currently using the services of the virtual machines.

The balancer then continuously scans these types of queue and the list of VMs. If there is a virtual machine available that can handle request of the node/client, the virtual machine is allocated to that request. If there is a virtual machine that is free and there is another virtual machine that needs to be freed of the load, then the balancer distributes some of the tasks of that virtual machine to the free one. So that it reduce the overhead of the former virtual machine [42]. This algorithm is also known as equally spread current execution load balancing algorithm. Because of it uses active monitoring load balancer for equally spreading the execution of loads on any different virtual machines in cloud computing environment [42].

For each virtual machine the active monitoring load balancing algorithm maintains information and the number of requests currently allocated to which virtual machine.

At the time of the requests/tasks to allocate a new virtual machine arrives the load balancer identifies the least loaded virtual machine. Here if there are more than one virtual machine the first identified is selected for the execution purpose.

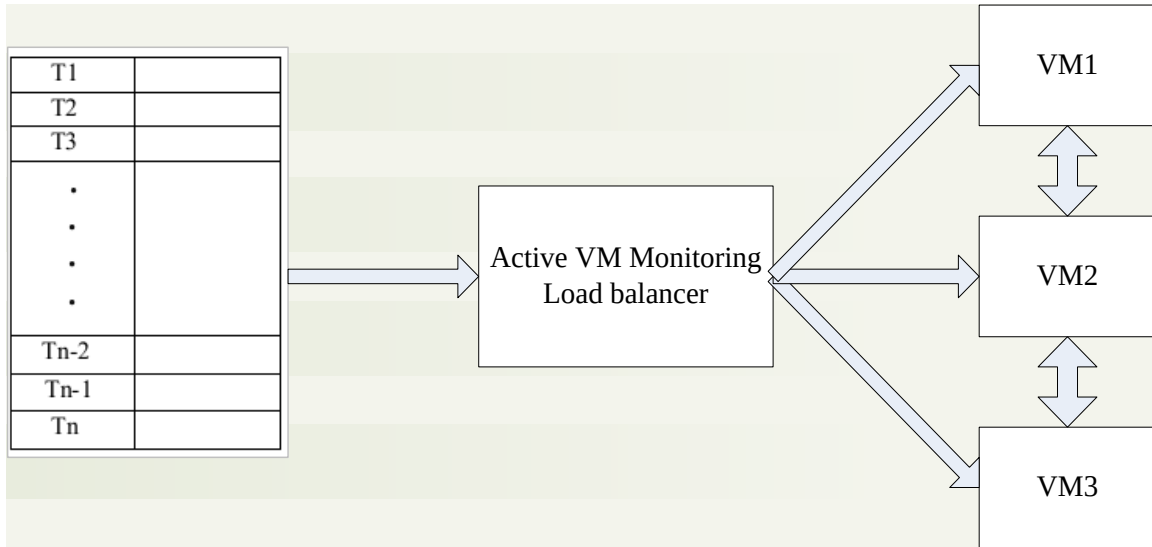


Figure 6: Active monitoring algorithm [44]

This active monitoring load balancing algorithm requires a load balancer which monitors the tasks/requests which are asked for execution. In this algorithm the task of the load balancer is to queue up the tasks/requests and hand over them to that of the different VMs. For that of the new tasks/requests the load balancer looks over the queue frequently and then allots them to the list of free virtual server. The load balancer also maintains the list of tasks/requests allotted to virtual servers, which is useful to identify that which VMs are free and need to be allotted with new tasks/requests. According to [32] the active monitoring load balancing technique plays an important role to have an equal amount of tasks/requests on all the available VMs. Depending on the above explanation the steps that involved in active monitoring load balancing algorithm as follows:

1. In this active monitoring load balancing algorithm initially all the virtual machine have zero (0) allocation. And the virtual machine index table updated by active virtual machine load balancer.
2. At this step when the request comes from the clients/users the data center controller requests for the assignation of a new virtual machine. And then check from the available virtual machine the least loaded one in the virtual machine allocation table.
3. In this stage among the available virtual machine the least loaded is more than one the first identified is selected.

4. After all this steps completed the identity number of the virtual machine sent from AMLB to data center controller.
5. Then the data center controller send the call to the virtual machine which identified by ID.
6. Here also the data center controller responds this process to the active virtual machine load balancer.
7. In this step the active virtual machine load balancer updates the allocation table and increase the allocation count.
8. After the virtual machine finishes its process and then the data center controller receives that of the corresponding cloudlet.
9. The allocation count increase by one (1).
10. The processes continue from step two (2).

2.4.3 Throttled Load Balancing Algorithm

As [26] explained about the throttled load balancing algorithm *“it is a static load balancing algorithm. Here it first checks the index values of all the virtual machine in the system”*.

After that the request is sent where the load balancer parses a table for the allocation of the resources in the system. And it assigns the request to a particular load balancer which passes or responds reverse the request to the requester and updates the allocation policy. After the successful allocation of the system the whole process for the de-allocation of the system also starts. The author mechanism provides a higher amount of resource sharing and allocation in the system resulting in the higher performance and utilization. This algorithm ensures that pre-defined numbers of cloudlets are allocated to a single VM at any given time. If there are more request groups are present then the number of available VM's at datacenter allocate incoming request, otherwise queues until the next VM becomes available.

In this section, we briefly summarized about the load balancing algorithms used in the cloud computing environment. The main aim is on the efficient utilization of the virtual machines and balancing the virtual machines with the incoming request. As [26], *“Load balancing is defined as a process of making effective resource utilization by reassigning the total load to the individual nodes of the collective system”*.

In order to improve the response time and to achieve efficient usages of the available VM have been attempts to make. The proposed algorithm selecting a VM for processing client's request where, VM at first index is initially selected depending upon the state of the VM. If the VM is available, it is assigned with the request and id of VM is returned to the data center, else “-1” (minus one) is returned. When the next request arrives, the VM at index next to already assigned VM is chosen depending on the state of VM.

And follows the above step, unlikely of the throttled algorithm, where the time of the index table is parsed from the first index. Every time the data center queries load balancer for allocation of the VM.

The [27] have developed throttled algorithm which is completely based on virtual machines. Here the client first requests the load balancer to check the right virtual machine. Which access that load easily and perform the operations which is given by the client [27]. In this algorithm the client first requests the load balancer to find out a suitable virtual machine to perform the required operation.

As [46] explained about the throttled load balancer (TLB) algorithm, it ensures that pre-defined numbers of cloudlets are allocated to a single virtual machine at any given time. In throttled load balancing algorithm, if there are more requests are present than the number of available virtual machines the data center controller allocates the incoming requests in queue basis until the next virtual machine becomes available. And also in this algorithm at any given time, only the certain quantity of the internet cloudlets will be distributed to the particular VM.

If the VM receives more number of requests than it can handle and whenever any VM becomes available at any time, the requests will be queued up. And then the VM gets all the queued up requests for processing [32].

According to [48, 49] “*this algorithm is a dynamic load balancing algorithm*”. It also plays an important role for load balancing in the case of the VM to be used. Here we first check that of the VM index values in the system.

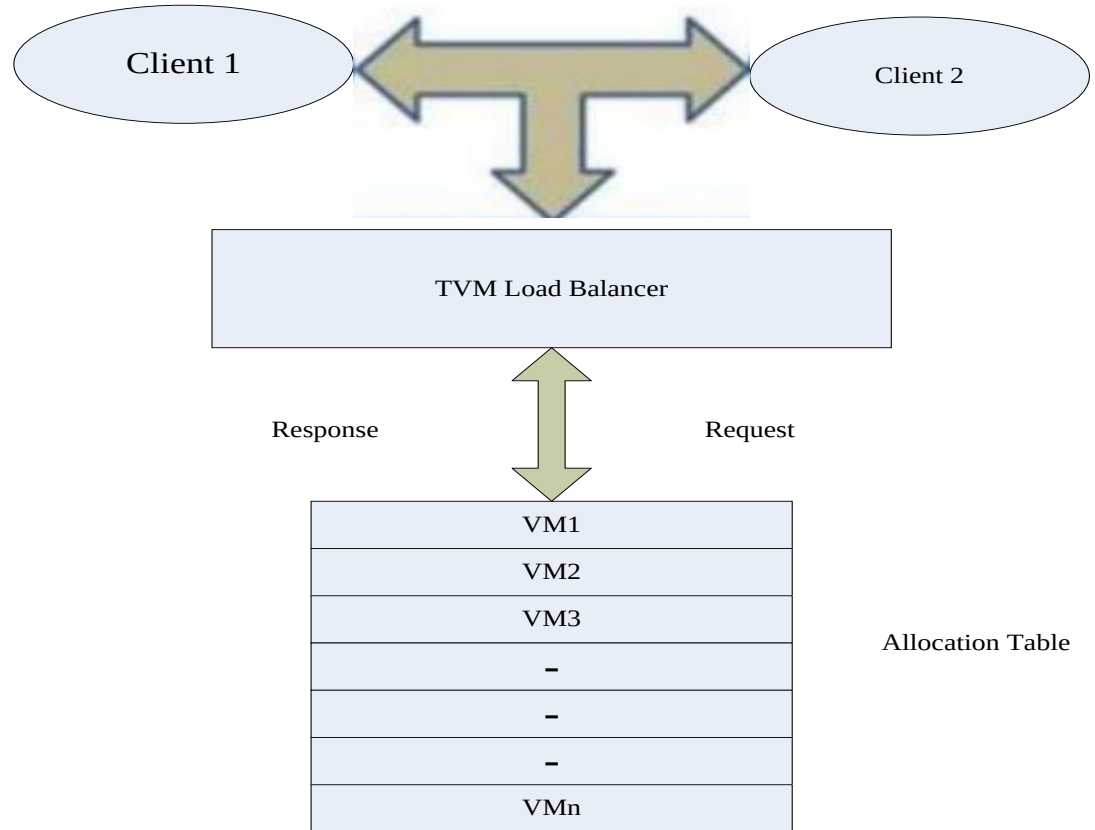


Figure 7: Throttled load balancing algorithm [4]

In the present work, considering of active VM load balancer and proposed VM assign algorithm for comparison. The main focus is to distribute the load efficiently on the available virtual machines and ensuring that under or over utilization of the resources/virtual machines would not occur in the cloud system.

The algorithm is the virtual machine assignation of load balancer:

Input: Number of incoming requests (assume, x) $x_1, x_2, x_3 \dots x_n$

Available VM (assume, y) $y_1, y_2, y_3 \dots y_n$

Output: All incoming requests, $x_1, x_2, x_3 \dots x_n$ are allocated least loaded virtual machine among the available virtual machines.

1. Initially all the VM's have no allocations.

2. VM-assign load balancer maintains the index/assign table of VMs which has number of requests currently allocated to each VM.
3. When requests arrive at the data center and then it passes to that of the load balancer.
4. Index table is parsed and least loaded VM is selected for execution.

In case i:

- A. Check whether that of the chosen least loaded VM is used immediately in the last iteration.

If YES

Go to the step four to find the next least VM.

If NO

The least loaded VM is chosen for execution.

5. The VM-assign load balancer returns the VM identification to the DC.
6. Then the request is assigned to the VM and the DC notifies the VM-assign load balancer about the allocation.
7. The VM-assign load balancer updates the requests which hold by each VM.
8. When the VM finishes the request processing, DC receives the response.
9. DC notifies the VM-assign load balancer for the VM de-allocation and the VM-assign load balancer updates the table.
10. Repeat from step 2 (two) for the next request.

A. Static Load Balancing Algorithm

Depend on the types of the general load balancing algorithm, static load balancing algorithm is the one which distributes the workload based strictly on a fixed set of rules such as input workload.

As [14] explained “*the load does not depend on the current state of the system, but it requires knowledge about the application and resources of the system*”.

B. Dynamic Load Balancing Algorithm

The dynamic load balancing algorithms are more flexible than that of the static algorithm and don't rely on prior knowledge, but depends on current state of the system [14].

[41], *"In a distributed system, dynamic load balancing has two different ways: distributed and non-distributed. In the distributed, the algorithm is executed by all nodes present in the system and among these servers the task of load balancing is shared. The interaction among nodes to achieve load balancing can take two forms: cooperative and non-cooperative. In the first one, the nodes works side-by-side to achieve a common objective which means to improve the overall response time, etc. In the non-distributed, each node works independently toward a goal local to it". The requested time for both the techniques applied are same which means there is no effect in request time of data centers after changing the algorithms"*.

In dynamic load balancing algorithm using the help of experiments to find out the cost analysis for that of each algorithm for virtual machine usage per hour is calculated. And the cost analysis for each algorithm is not same for two algorithms.

2.5 Optimization of Resources in Cloud Computing Using Effective LBAs

According to [28] descriptions, in cloud computing the load balancing is a methodology. It is used to distribute tasks across multiple machines to get resource utilization with less response time. In this paper to choose good load balancer at the initial phase is imperative to get success of complex implementations. So that studied about the different types of load balancing algorithm and then proposed an algorithm. The proposed algorithm used to assign least virtual machine method to distribute tasks across multiple machines to get resource utilizations.

To do this for the algorithm implementations purpose used a cloudsim simulation. The algorithm is as follows:

Least virtual machine assign algorithm ()

{

Input: Task= $T_1, T_2, T_3, \dots, T_n$

Virtual machines have no allocations (for all).

Old VM₁, Old VM₂, Old VM₃, ... Old VM_n=0;

New VM₁, New VM₂, ... New VM_n=0;

If (Old VM₁=1)

```
{  
    New VM1 += 1;  
}
```

Else if

```
{  
    New VM1 += 1;  
}
```

Selected VM = least virtual machine of (New VM₁, New VM₂, ... New VM_n)

If (Selected Virtual machine is free)

```
{  
    Selected VM = Request (T1, T2, T3, ... Tn)  
}
```

Else

```
{  
    Least virtual machine = Selected virtual machine  
}
```

Go to Selected virtual machine.

2.6 A Comparative Study of Load Balancing Algorithms in Cloud Computing

As cloud computing has been growing rapidly and many clients all over the world are demanding more services and better results. The load balancing is an essential and important research area. For the case of allocating client's request to available remote nodes many algorithms have been developed.

According to [29] the efficient load balancing ensures efficient resource utilization of resources to customers on demand basis and enhanced the overall performance of the cloud.

As [29] explained about the cloud computing, *“it is the fastest developed technology in the IT industry and a new delivery method for the services on pay per use basis”*. Studying about the different types of load balancing algorithms are very important [29].

To understand and to schedule the incoming requests with minimum response time, efficient resource utilization and at the same time resources should not be underutilized. The main objective of load balancing is to increase client satisfaction, to increase resource utilization, to maximize the performance of the cloud system.

As a final this paper briefly discussed about the existing load balancing techniques in cloud computing. And further compares them based on various parameters like data processing time and response time. The paper analyzes the results based on the existing round robin and throttled scheduling algorithms. For the two algorithms the requested time are same.

Based on the cost of usage the throttled load balancing algorithm is better than the round robin load balancing algorithm on cloud data centers.

2.7 Efficient Load Balancing Algorithm in Cloud Environment

In efficient load balancing algorithm in cloud computing environment presents *“benchmarks for the algorithms to show their capabilities, benchmark workload is synthetic to highlight many aspects of balancing load correctly”* [30].

From the benchmarks and by analyzing the different algorithms show a simple algorithm that is platform independent, easy to implement and has the best performance of any of the algorithms implemented [5, 6, 30].

2.7.1 Load Algorithm Active VM Load Balancing [31].

Step1: Find the next available virtual machine.

Step2: Check for all current allocation count is less than max length of virtual machine list allocates the virtual machine.

Step3: Among existing virtual machines if, available VM is not allocated create a new one.

Step 4: In this step count the active load on each virtual machine.

Step5: Return the ID of that virtual machine which is having least loaded.

The END

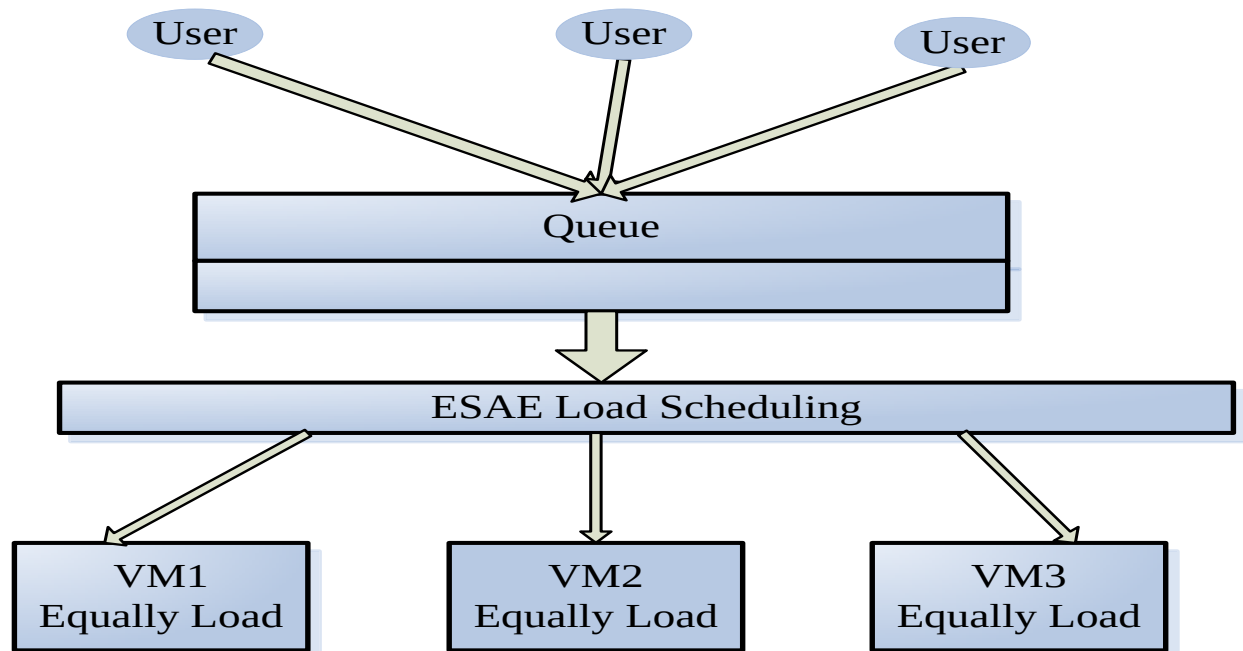


Figure 8: Equally spread active execution load to the cloud system

As the paper aims is towards the development of enhanced strategies through improved job scheduling and resource allocation techniques for overcoming the above stated issues [32]. Here equal spread current execution load balancing algorithm dynamically allocates the resources to the job in queue leading and reduced cost in data transfer and virtual machine formation. Comparisons of various load balancing algorithms are made with respect to overall time and cost [32]. Cost and time are the key challenge of every IT engineer to develop products that can enhance the business performance in the cloud based IT sectors.

According to [31, 33], the various VM load balancing algorithms are analyzed. Here the new load balancing algorithm implemented for an IaaS framework in simulated cloud computing environment i.e. "*Weighted Active Monitoring Load Balancing Algorithm*". And also in the VM load balancing algorithm the approach in which an individual application services is assigned different amount of the available processing power of VMs.

This is because in the real world, it's not necessary all the VMs in a datacenter has fixed amount of processing powers, but it can vary with different computing nodes at different ends.

And then to these VMs of different processing powers, the tasks/requests are assigned or allocated to the most powerful VM and then to the lowest and so on. They are given the required priority weights. Hence, the performance parameters such as overall response time and data processing time are optimized.

The [11, 29] have defined the problem as the random arrival of load in such an environment can cause some server to be heavily loaded, while other server is idle or only lightly loaded. Equally load distributing improves performance by transferring load from heavily loaded server. In the algorithm proposed by the author the active VM load balancer finds that of the next available VM. The load balancer checks that for all current allocation count is less than maximum length of VM list in order to allocate the VM. If available VM is not allocated then a new one is created.

And at last active load is counted on each VM and ID of those VM is returned which is having least load. The simulation result shows the overall time, the cost results and comparison of load balancing algorithms. ESCE load balancing provide better results as compared to round robin on closet data center and optimise response time [11].

A load balancer algorithm that controls the flow of pay load based on the safety thresholds, which may be static or dynamic in nature, depending on the available machines and bandwidth are developed by [34, 35]. And analyzed the existing algorithms of load balancing such as: round robin, throttled, equally spread and biased random sampling. And have proposed a new algorithm from the existing load balancing approach. To do this by decreasing the overall requesting time and processing time as compared to the existing algorithms. In the proposed approach the [34] was worked on virtual machine migration policy by giving priority to VM's on the basis of resources available to it. Firstly, the algorithm checks that whether the CPU utilization of the virtual machine is equal to, greater than or less than 80%. The value has been chosen in order to prevent the machine from being overloaded. It is clear from the results that the proposed method in the paper is able to balance the load to a greater extent than the analyzed algorithms. This load balancing algorithm aims at providing dynamic, on-demand, balance of resources available to the resources required to accomplish the task.

Since, this algorithm involves reallocation of resources involving virtual machine and data centers. Therefore, every time whenever re-allocation occur re-authentication of the resource allocation is also conducted along with re-allocation of load balancing. So, this algorithm also provides better security, by focusing on the concept of not disclosing the personal details of the user to cloud provider.

The scheduling strategy should be developed for multiple tasks proposed by [28]. In cloud computing, processing is done on remote computer: hence there are more chances of errors, due to the undetermined latency and lose control over computing node.

A computing node is selected for computation on the basis of its reliability and can be removed, if does not perform well for applications. *“Reliability assessment algorithm is applied on each node (virtual machine) one by one”*. Initially reliability of a node is set to 1 (one). There is an adaptability factor “*n*”, which controls the reliability assessment. The value of “*n*” is always greater than 0 (zero) [28].

The algorithm takes input of three factors RF, minimum reliability and maximum reliability from configuration file.

RF is a reliability factor which increases or decreases the reliability of the node. Min reliability is the minimum reliability level. If a node reaches to this level, it is stopped to perform further operations.

2.8 Related Work

In cloud computing environment in order to balance the load a number of load balancing algorithms have been developed. Now a days many researchers concentrate on the load balancing algorithm to maintain load in cloud computing environment.

This study focuses to solve the complexity of the active monitoring load balancing algorithm and to describe about the utilization concept of the algorithm. As [36] discussed about the round robin load balancing algorithm the request is sent to the node having least number of connections, and because of this at some point of time, some node may be heavily loaded and other remain idle. But here the drawback of round robin load balancing algorithm is, it simply works on time slicing.

In this algorithm among the available virtual machines the data center controller randomly picked the virtual machine and then assigns the first request to that virtual machine. And also allocates request in circular order. Here after assigns the request to the picked virtual machine then the virtual machine moved to the end of the list. As a final the author concluded that the advantage of this algorithm doesn't require inter-process communication and the running time of any process is not known to execution.

In [3, 37] active monitoring load balancing algorithm, it checks all virtual machine in sequence for its availability to allocate resource on a free virtual machine [36]. And also has to wait for its next iteration and then if the request is complete, it allows other process of the virtual machine based on the algorithm. The active monitoring is dynamic in nature. Which means it keeps information about each virtual machine and number of requests assigns to the virtual machines.

In throttled load balancing algorithm according to [26] if more request groups are present then the DCC allocates to the VM. Otherwise puts on queue until the next VM becomes available. To improve the response time of the throttled load balancing algorithm among the state of the VM list selects the first index. Here if it is ok the algorithm assigned the request to the virtual machine and returned the VM id to the data center controller. Otherwise minus one returned.

At the time of next request comes the virtual machine at index next is chosen among the available virtual machines. The throttled load balancing is static load balancing algorithm. So that it first check the index value of all the virtual machines in the system.

Based on [14] research paper the classification of load balancing algorithm in cloud computing have been proposed for different load balancing algorithms, such as; *“random sampling, active clustering, honeybee-based load balancing technique, throttled load balancer and active monitoring load balancer”*. The main purpose was to achieve high throughput and minimum response time. In the paper the researcher finally discussed about the load balancing was a main task in cloud computing for efficient utilization of resources.

And also the main goal of load balancing is to achieve higher client satisfaction, maximize resource utilization and increase the performance of the cloud system thereby reduction in the energy consumption. The performances of two algorithms round robin and throttled have been discussed in the paper. Here throttled load balancing algorithm reduces the cost of usage. Therefore the researcher recommended that, *“the throttled load balancing technique is more efficient in terms of cost for load balancing on cloud data centers”*.

In the research paper of [36] *“A New Static Load Balancing Algorithm in Cloud Computing”* is proposed. The proposed algorithm is using the concept of both active monitoring load balancing algorithm and throttled load balancing algorithm. In the paper the detailed design, pseudo code and implementation of algorithm are also presented. The overall response time and data center processing time results are obtained and compared with the results of throttled load balancing algorithm. This comparison is done after implementing and analyzing each of the existing algorithms discussed in this paper. And at last found that throttled load balancing algorithm is best among all the existing algorithms.

As [29] discussed about the load balancing algorithms for essential operations in cloud virtual environment and cloud computing. And also briefly discussed about the existing load balancing techniques in cloud computing and further compares them based on various parameters, like data processing time and response time. The paper analyzed the result based on existing round robin and throttled scheduling algorithms.

In this paper the main problem was to schedule the incoming requests in a way so with minimum response time, efficient resource utilization and at the same time resources should not be underutilized.

As [5, 6, 11] discussed about an approach for scheduling algorithms that can maintain the load balancing. And also provides better improved strategies through efficient job scheduling and modified resource allocation techniques.

According to [31, 33] the author has concluded that *“the proposed load balancing algorithm has optimized the given performance parameters such as response time and data processing time, giving an efficient VM load balancing algorithm”*.

In [2, 29] have been provided the solution based on scheduling algorithm ESCE (Equally Spread Current Execution) load.

The author has then compared the performance result of ESCE with round robin scheduling to estimate response time and processing time. As a final concluded after simulating the algorithm using a cloudsim simulator, the ESCE algorithm dynamically allocates the resources to the job in queue leading reduced cost in data transfer and virtual machine format icon.

The researcher solves this determines problem and tries to give a fault tolerant model for the cloud computing [28]. As a final concluded that *“the paper with the simulation results showing that the proposed fault tolerant technique works better than other similar fault tolerant techniques”*. The result of the simulations shows that for all the processors in the system and in the network approximately equal amount of work at any given time [28].

As a final we concluded about the related works which are done at load balancing algorithm in cloud computing environment. The reasons behind to study about the proposed load balancing algorithm in cloud computing environment as follows:

In cloud computing system, load balancing is really a challenge now because; always a distribution solution is required. And also when we see the different types of load balancing algorithms in cloud computing environment, as cloud computing has been growing rapidly and many clients all over the world are demanding more services and better results.

But to get this there was a challenge to schedule incoming requests in a way so with minimum response time, efficient resource utilization and others. In round robin load balancing algorithm, among the available virtual machines the DCC randomly picked the virtual machine and assign the first request to that virtual machines. And also assigns requests in circular order. In this technique at some point of time some node may be heavily loaded and others remain idle. And the other drawback of this algorithm is it simply works in time slicing.

On the other hand in throttled load balancing algorithm, the algorithm starts searching from last allocated virtual machine to the n^{th} virtual machines. But, in this algorithm there is still a problem which is, it doesn't utilizes those virtual machines which become free after the execution of the request.

The active monitoring load balancing algorithm has also the same problem and also additionally if, the number of request increases on a virtual machine the complexity also increases.

CHAPTER THREE

3. METHODOLOGY OF THE RESEARCH

The main aim of this chapter was to design the methodology to carry out the study, which included: determining design strategy for the study including a conceptual framework, defining the deliverables, and articulating the methodology. The requirements to this study were: developing the knowledge base about the state of the problem and the importance of its solution to define the problem; developing the knowledge base about the state of the problem and current solutions of the problem to identify the requirements; the knowledge of theory that brought to bear in a solution to design the proposed load balancing algorithm in cloud computing; and developing evaluation source tool to measure the performance of the resulting of the load balancing algorithm [39].

Research methodology solves the research problems by logically adopting various steps. Methodology helps us to understand not only the products of scientific inquiry but the process itself.

Research methodology is a way to systematically solve the research problem. It may be understood as a science of studying how research is done scientifically. In it we study the various steps that are generally used in studying of the research problem along with the logic behind the research problem. It is necessary for us to know about the research methods/techniques and the methodology [40].

3.1 Research Design

A research project is conducting using research design. It specifies the details of the procedures necessary for obtaining the information needed to structure to solve the research problems. And also it is the specification of procedures for collecting and analyzing the data which is necessary to identify the problem. Such that: the difference between the cost of obtaining various levels of accuracy and the expected value of the information associated with each level of accuracy is maximized.

Designing of the research is done mainly to solve the problem of getting the various stages of the research under control. This control factor is very important for us (researcher) during any of the research operation. Preparation of the design for the research forms a very critical stage in the process of carrying out a research project. In our study the research design plays an important role to integrate the different components of the study in a coherent and logical way. Thereby, ensuring we should effectively address the research problems; it constitutes the blueprint for the collection, measurement, and analysis of data.

In our research the study is focus on the descriptive research type. This research type concerned with describing the description of functions. And describes systematically a situation, problem, phenomenon, service or program, or provides information. It includes the fact-finding of enquiries of different kinds the state of affairs as it exists at present. And also the descriptive research refers to research that supply an accurate description of characteristics of a particular individual situation, or group.

In our study we also follow the qualitative research method, which contributes to the discovery and theory building. And also this research with respect to a cloud computing based system attempted to involve quality in having insight into problems.

I. Problem identification and motivation

In this phase we define the state of the problem and the importance of its solution. To do this by defining the specific research problem of the load balancing algorithm and justify the value of a solution analysis of literatures on a load balancing algorithm adoption in cloud computing environment.

II. Define the objectives for a solution

Depending on the knowledge of the state of problems we infer the objectives of a solution and knowledge of what is possible with the cloud computing requirements. The qualitative objectives are used, which described how a proposed algorithm is expected in order to support solutions to the problems.

III. Design and Development

In the design and development phase a particular processes are selected. And then the data confidentiality and availability strategies are completed. After determining the algorithm architecture and its functionality, the proposed collaborative load balancing algorithm in cloud computing with the implementation plan to deploy application on the cloud could be developed.

IV. Evaluation

The evaluation phase used to observe and measure how well the algorithm supports a solution to the problem. This activity involves comparing the objectives of a solution to actual observed results from use of the cloud computing algorithm in the demonstration. It requires the knowledge of analysis techniques. It depends on the nature of the problem of active monitoring load balancing algorithm at virtual machine level in cloud environment.

The study took the qualitative evaluation analysis techniques which include: a comparison of the existing load balancing algorithms with the proposed load balancing algorithm functionality with the solution objectives. Qualitative performance measures, such as, extents of overall response time and the closest data center processing time of the existing and the modified load balancing algorithm in cloud computing environment.

3.2 Data Source

For the study purpose in the research both primary and secondary data are used. We need to consider the sources on which to base and confirm our research and findings. A Data Source is an abstract representation of a live set of data that presents a common predictable application program interface (API) for other objects to interact with [39].

And also it is the source of the data. It can be a file, a particular database on a DBMS. The data might be located on the same computer as the program, or on another computer somewhere on the network. In this study the data source is the international network. That means the data collected from the internet which is the twitter user's dataset in six geographical locations in the real world. The twitter users of the sample are shown for the following table for the study purpose.

Table 1: Source: (John Ribeiro , May 27,2014)

Main Continent	Africa	Asia	Europe	North America	Oceania	South America
Year	2014	2014	2014	2014	2014	2014
Number of twitter users (in million)	11.8	32.8	419,965,132	512,241,876	107.7	415,412,166

“In the organization the data sources can be designated as structured or unstructured data for classification. The term structured data refers to the data which is identifiable because of it is organized in a structure”.

(Rutger Kortleven, Ian Hopkinson (20113))

The database is the most common form of structured data or structured data records (SDR). In the database the specific information is stored based on a methodology of columns and rows. Within the content the structured data can be searchable by a data type. And can be understood by electronic machines and is efficiently organized for human readers. In contrast, unstructured data refers to the data that has no identifiable structure or common form of structured data.

Within a dataset, like images, videos, email, documents and text are all considered to be that of unstructured data. In an unstructured data each individual document may contain its own specific structure or formatting that depends on the software program used to create the data [39]. In unstructured data the dataset will not contain the same structure.

At the time of proposing the thesis, it differs in a number of aspects from one to another. But they do have some commonalities. One of the common aspects is the need to collect data to do research project. Data collection can be derived from a number of methods, which include interviews, focus groups, surveys, telephone interviews, field notes, questionnaires, and from various publications. But in this research we collect the data through action search.

Generally the data collection consists of either primary or secondary data.

A. Primary Data

The Data collection can be either primary or secondary data. The Primary data collected by the researcher to solve the research problems [40]. The primary data is a fresh data and collected for the first time. It is also known as the original data that has been collected from the original place for the different purpose. This type of data is generally a fresh and collected for the first time. The primary data plays an important role for current studies as well as for future studies.

The primary data are information collected by a researcher specifically for a research project. In other words, primary data are information that an organization must gather to be accessible to the public because, no one has compiled and published the information in a forum. And it is original in nature and directly related to the issue or problem and the current data. To do research we have expected to collect the primary data through various methods like interviews, surveys, questionnaires etc.

“Most of the time the original reports found in academic journals. And also the detailing of the research methodology used in-depth descriptions of the research. Based on the primary source of the information the findings of the discussions are also considered”.

(Galvan, J. L. (2013))

B. Secondary data

Secondary data is also data that is being reused in different context like data from book. It also involves less cost time and effort than that of the primary data. The secondary data can be collected by a party which is not related to the research study but for other purpose at any time in the past. In our research when we use these data for current research, it is secondary data.

Secondary data is use of information already collected by someone else. Secondary data may either be published data or unpublished data [40]. And it can be available in the written, typed or in electronic forms. As [40] explained *“Secondary data is use of information already collected by someone else”*.

The different types of secondary information sources are available to us in order to gather on an organization, potential product applications and the market place.

And variety of secondary information sources is available to the researcher gathering data on an industry, potential product applications and the market place. And also the secondary data plays an important role to gain the initial insight into the research problem. The secondary data can be classified based on its source which is either internal or external. When we see the internal data it is the secondary information which acquired within the organization where the research is being carried out. The second one is the external secondary data which is obtained from that of the outside sources.

A secondary source can be created by someone at later time than the event being researched and the one who created it did not experience the said event. That means the created document have not direct connection with the events. When we see the secondary source it is written about the primary sources and that provides non-original data or information. It also the interpretation based on the primary sources and the work of the other writer.

In the secondary source the authors develop their interpretation of events based on the primary sources. At the same time also take the advantages using the work of other historians using other secondary sources.

For example, the author of the load balancing algorithm in cloud computing textbook which we use in the university probably did not use too many primary sources; instead, the textbook authors rely on secondary sources written by other writers.

Generally the original documents of the primary sources are a creative work. Whereas the secondary sources develop the theories related to a topic by describing, interpreting, evaluating, explaining and the like. And also it is written after the fact within the hindsight [29].

3.3 Data Collection Techniques

The data collection is gathering of data for a particular purpose from various sources and which is used to explain about the process of the collecting data. The information or data that can comes from the range of the sources. To collect the primary data we use different types of techniques. For instance in our research we used googling or action search techniques to collect data from the internet.

Data collection is the process of gathering and measuring the information or data on objected variables in the established systematic fashion.

And also it enables a person or organization in order to answer relevant question and to evaluate the outcomes and make predictions about the future probabilities and trends.

In the research study the data collection is defined as the systematic collection, analysis and interpretation of the necessary data to design implement and evaluate. It is also used to develop effective strategies.

3.4 Issues of Reliability and Validity

In the research the validity and reliability are the important concepts that capture the measurement properties of a survey, questionnaire or another type of measure. The everyday use of these terms provides a sense of what they mean. The concept of validation refers to the determination of the correctness of the system with respect to user needs and requirements [40].

In our research there are numbers of validation criteria which include comparisons of service broker policy with overall response time, overall closest data center processing time and others. The quality of research defined depending on reliability and validity and measures using the CloudAnalyst tool to supposed to test and perform as it is designed. So that the validity plays an important role to reflect true differences among those being tested in load balancing algorithm performance [41].

And the findings truly represent the phenomenon that we are claiming to measure. In the same way our study is reliable because when we test the result for the second time it should be yield the same result.

As a final, in our study the outcomes are measured by considering the objectives using qualitative measurement analysis techniques using the CloudAnalyst tool. And described how a proposed load balancing algorithm in a cloud computing environment is expected to support solutions to the problems.

3.5 Sampling Techniques

Sampling is selecting a suitable sample or a representative part of a population for the purpose of determining parameters or characteristics of the whole population [39]. We used a smaller number of items/parts of among the larger population to make conclusion about the entire population.

In this study the sampling population of the research includes any active enabling registered users to read and post messages in twitter of the selected six (6) regions that coincide with the six (6) main continents in the world. The other main entities such as UBs (user bases) and data centers belong to one of these regions. This geographical grouping is used to maintain a level of realistic simplicity for the large scaled simulation being attempted in the CloudAnalyst tool.

In this study based on the sampling techniques as we mentioned in the data source out of the six regions, which coincide with six main continents in the world by taking sample of twitter users in these main continents. Assume here one UB represents as thousands of peoples in the regions.

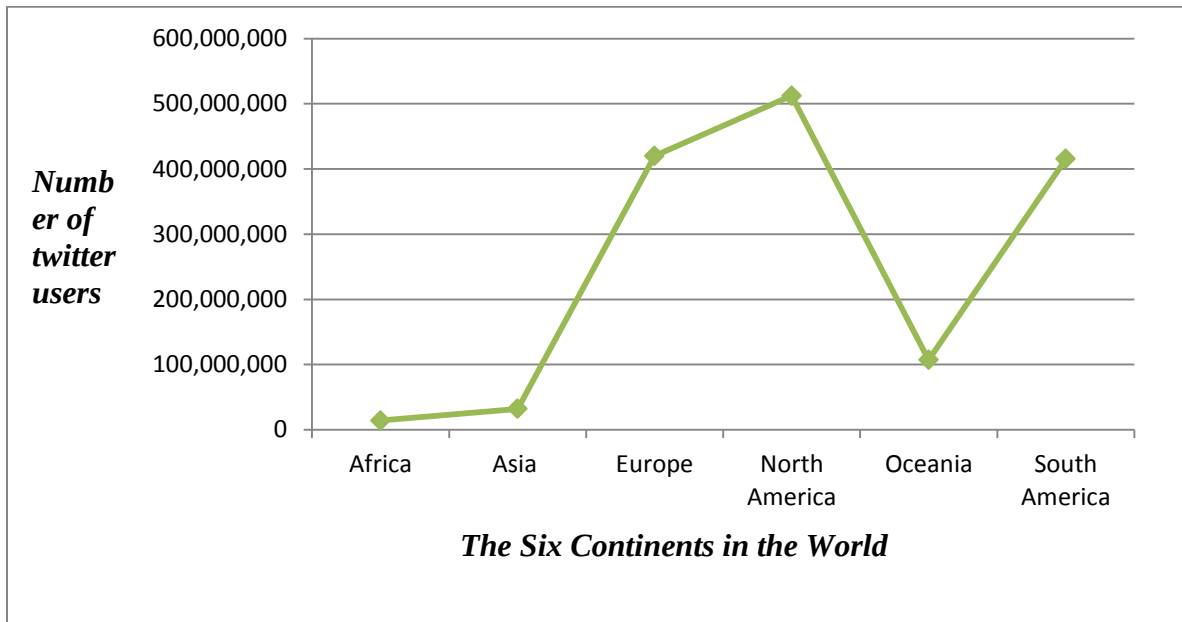


Figure 9: Twitter users in 2014. Source: (John Ribeiro, May 27, 2014)

3.6 Open Source Cloud Computing Tools

3.6.1 CloudAnalyst

In order to analyze the above discussed algorithms we used the tool called CloudAnalyst. Basically CloudAnalyst is a cloudsims based graphical user interface tool used for modeling and analysis of large scale load balancing algorithms in cloud computing environment.

The objective of this tool is to support evaluation of social network tools according to geographic distribution of customers and data centers. It is also a graphical simulation tool based on the cloudsims for modeling and analyzing behavior of load balancing algorithm in cloud computing environment. And also supports visual modeling and simulation of large-scale applications that are deployed on cloud infrastructures algorithms [6].

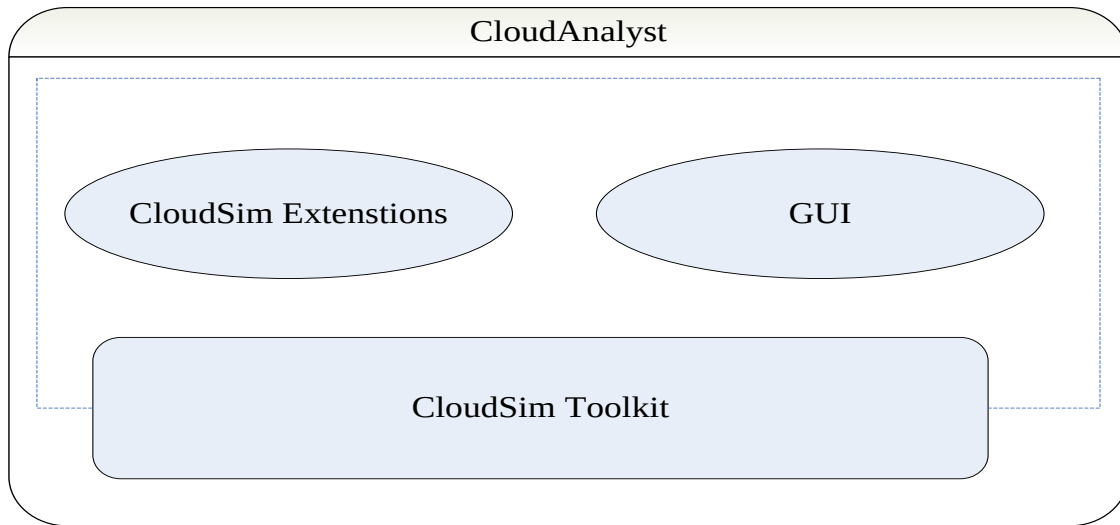


Figure 10: Architecture of CloudAnalyst simulator [27].

It also separates simulation set up environment exercise and supports to focus on the parameters used for simulation purposes rather than the programming technicalities [42]. The graphical user interface simulation output results of CloudAnalyst enables the results to be examined more straight forwardly with more proficiently. For a given load balancing algorithm in cloud computing environment also quickly helping in stressing out any problems dealing with the accuracy and performance of simulation.

Communities of customers and data centers supporting the social networks are characterized. While using the social network application and load on the data center, their location and parameters are obtained/logged [29].

The CloudAnalyst allows setting location of users, number of users and number of requests per user per hour. And also allows setting the location of the data centers, number of virtual machines, number of processors, amount of storage, network bandwidth and other necessary parameters which are used in the CloudAnalyst tool to test the experiment.

3.6.2 Cloudsim

In our research to analyze the problem of active monitoring load balancing algorithm at the time of the number of requests increases on a virtual machine we used the simulation tool cloudsim. To run the simulation we used java on IDE 7.2.1 NetBeans.

Cloudsim is a library in cloud computing environment for the simulation of cloud scenarios. It also provided that of the essential classes in order to describe data centers, VM, computational resources, the application, users and policies for the management of various parts of the system [27, 43].

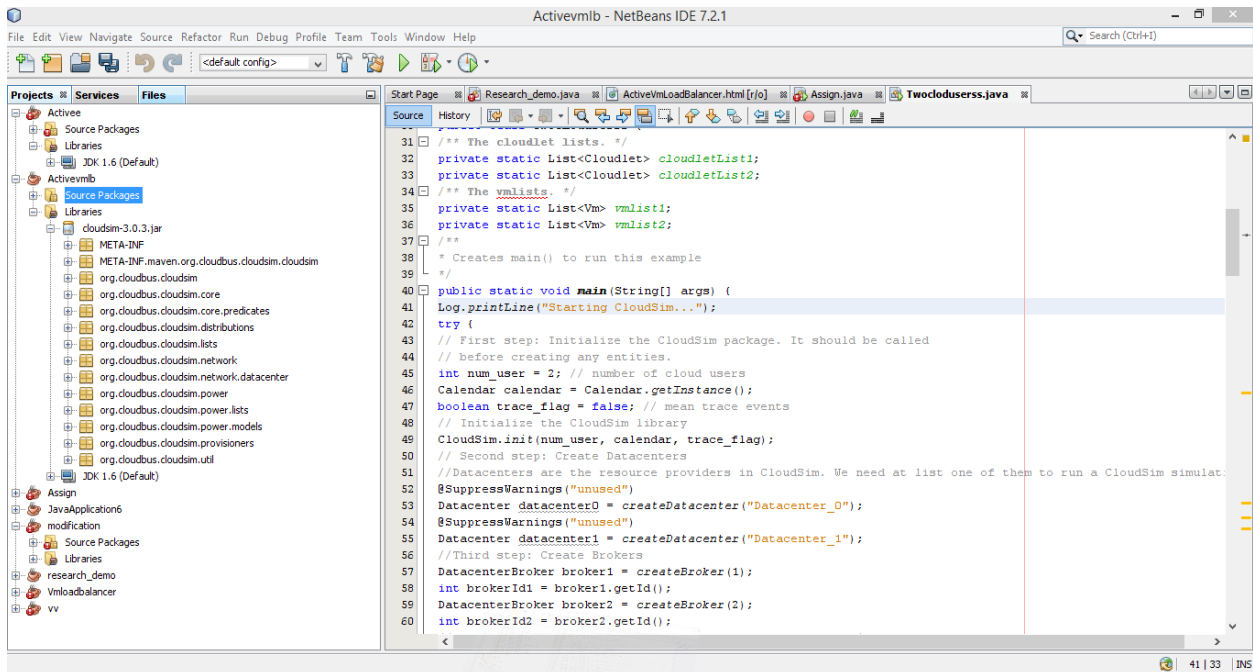


Figure 11: VM Load Balancer Using Cloudsim Library

3.7 Evaluation Approach

Based on the cloudsim open source tool the evaluation approach of the load balancing algorithms are as follows:

In existing load balancing algorithm when number of cloud users increases to access cloud resources the complexity of the load balancing algorithm also increases. But in the modified active monitoring load balancing algorithm among the available virtual machines when number of cloud users increases than that of the virtual machines, put the task to buffer until the virtual machine release some task. Based on this we can reduce the complexity of the algorithm and increased the performance of the algorithm in general. And also to select the least loaded virtual machine in order to assign request to the selected virtual machine for the execution purpose. On the same way at the time of assigning request to the selected virtual machine, it tells us the starting time, finishing time and others. This is used to understand about the response time and processing time of the algorithms.

CHAPTER FOUR

4. DESIGNING THE MODIFIED ACTIVE MONITORING LOAD BALANCING ALGORITHM

4.1 Proposed Using Load Balancing Algorithm VM-Assign

In cloud computing environment, the active monitoring load balancing algorithm focuses to find out that of the least loaded virtual machine from the existing available virtual machines in load balancing algorithm process.

Basically the load balancing algorithm sit in a place between the clients and the server in order to provide the services to the clients that of the client wants to use. In cloud computing, load balancing plays an important role to distribute workloads across multiple computing resources [4].

In this modified active monitoring load balancing algorithm at virtual machine level in cloud environment process, how the client incoming requests are assigned intelligently at virtual machine at the time of the least loaded virtual machine not equal to the last virtual machine.

In the figure bellow shows the general layouts of the modified active monitoring load balancing algorithm. How the user can use computing resources using the load balancing algorithm. To access resources without complexity and to show the utilization concept of the load balancing algorithm at virtual machine level in cloud environment.

The existing and modified active monitoring load balancing algorithm in cloud computing environment process as follows:

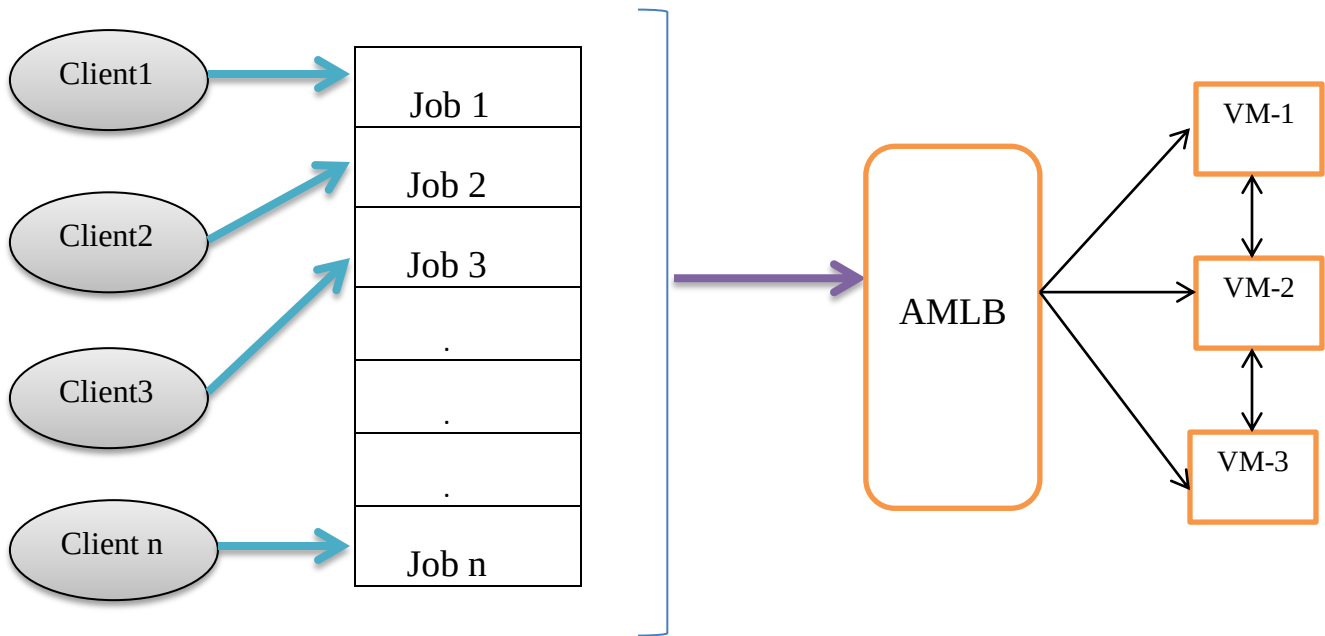


Figure 12: Flow of existing active monitoring load balancing algorithm

In this existing active monitoring load balancing algorithm, if number of cloud users increases on a virtual machine to access cloud resources the complexity of the active monitoring load balancing algorithm increases. This is because; if number of requests increases, there is no any mechanism in order to decrease the complexity of the algorithm.

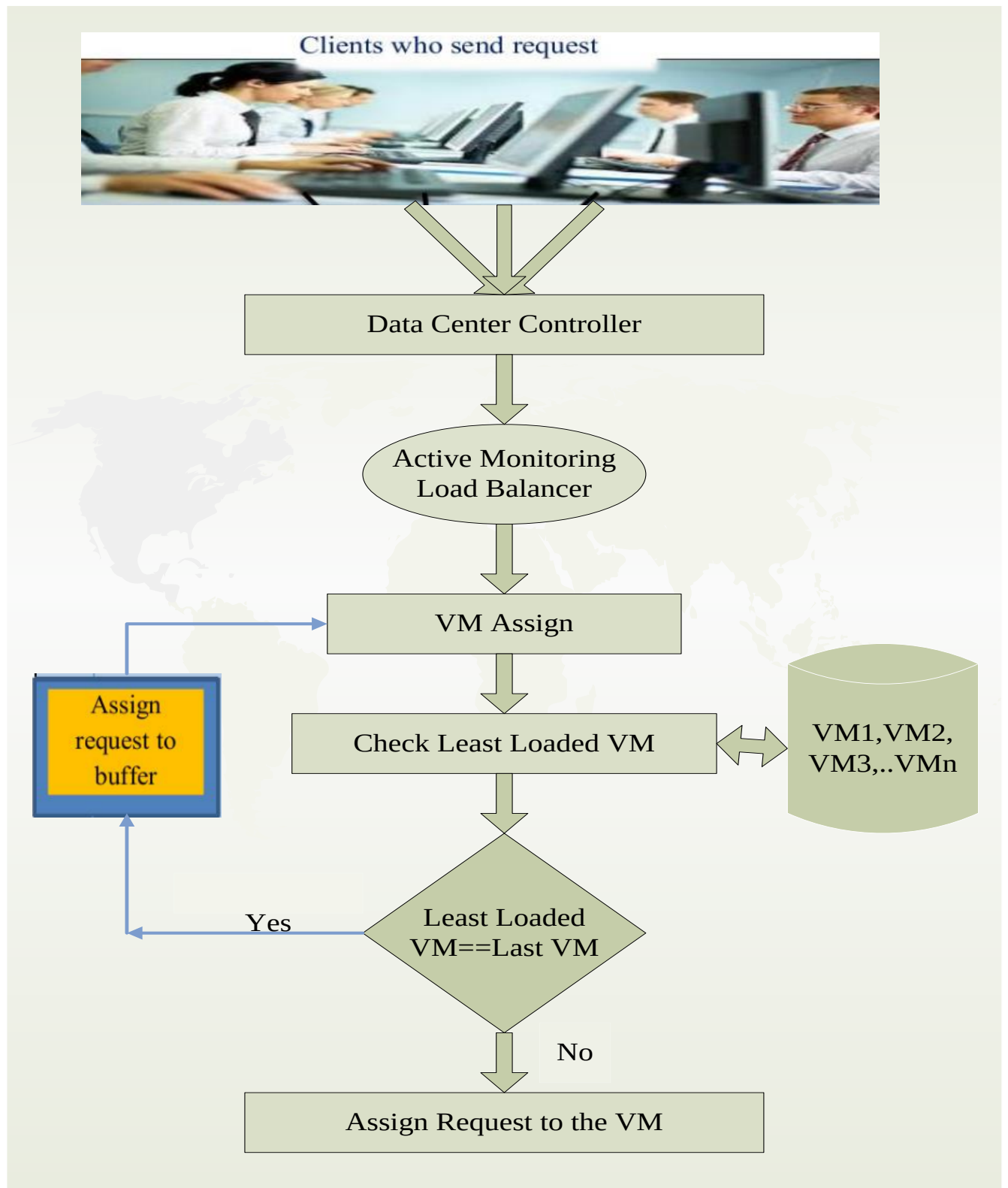


Figure 13: Flow of the Proposed VM-assign Algorithm

In the above figure the number of clients tries to access the cloud computing resources using the modified active monitoring load balancing algorithm at virtual machine level in cloud computing environment.

If the clients send their requests to the virtual machine through the data center controller, here at the time of the clients request arrives to the data center controller, then the DCC also sends to the active monitoring load balancer. And then the AMLB maintains the virtual machine table.

For the execution purpose the least loaded virtual machine is selected. Here among the available virtual machines, the selected one is the least loaded virtual machine. And doesn't used in the last iteration and then assign the tasks/requests to the selected virtual machine, otherwise assign the tasks/requests to buffer until the next least loaded virtual machine found.

The following steps illustrate the mechanisms for operating in the cloud computing environment:

Algorithm: VM-Assign AMLB.

Input: Incoming requests [assume request represents as R], ($R_1, R_2, R_3, \dots R_n$).

Available VM [VM represents as V], ($V_1, V_2, V_3, \dots V_n$).

Output: All incoming requests [R], ($R_1, R_2, R_3, \dots R_n$) are allocated to the least loaded VM among the available VMs.

Step1: At the starting time all the VMs have no allocation.

Step2: Number of incoming requests comes from number of clients to access computing resources.

Step3: At the time of the request arrives to the DCC then it sends to the AMLB.

Step4: Then the load balancer maintains the VMs index table to assign customer request to the virtual machine.

Step5: The least loaded VM is separated for execution purpose.

Case1: If the load balancer found the least loaded VM.

- a) Then check the selected least loaded VM which is used in the last iteration and reached in maximum limitation to accept any request.

If YES:

Assign the customer requests/tasks to buffer (which is temporarily stored data) and goes to step four to find next least loaded virtual machine.

- b) And then check from the available VM the next least loaded VM which is used in the last iteration and reached in maximum limitation to accept any request.

If NO:

Assign the request to the selected virtual machine.

Least loaded VM is chosen and assign request to the VM.

Step6: At this stage the load balancer replays the VM id to the data center controller.

Step7: Here the request is assigned to the VM and DCC tells to the LB about the assignation.

Step8: The load balancer updates the VM which holds the request.

Step9: At the time of VM finishes the processing the DC gets response.

Step10: The DC tells about the assignation to the load balancer for the virtual machine de-allocation.

Step11: The process repeats starting from step four for next requests/tasks.

CHAPTER FIVE

5. PRESENTATION, ANALYSIS AND INTERPRETATION OF DATA

This chapter comprises about the analysis, presentation and interpretation of the findings resulting from the study of the active monitoring load balancing algorithm.

Analysis

In the study to analyze means to break a topic or concept down into its parts in order to inspect and understand it. The steps involved in data analysis are a function of the type of data/information collected. The purpose of the assessment and the assessment questions will provide a structure for the organization of the data and a focus for the analysis.

In our research first analyzed the current status of the active monitoring load balancing algorithm from the selected load balancing algorithm in cloud computing. To do so the different types of load balancing algorithms are analyzed based on the CloudAnalyst tool in order to describe the utilization of the active monitoring load balancing algorithm. And also results are analyzed with respect to the efficient utilization of the virtual machines by avoiding the overloading condition. And then by comparing the existing active virtual machine load balancing with the modified one to understand the performance of the load balancing algorithms in cloud computing environment. Equal load distribution may improve resource utilization in cloud computing environment. This research study carefully investigated the possible benefits for any user who can use cloud computing in each day to day life.

Interpretation

In the research report the interpretation is the most critical aspect to answer the question “*so what*” in relation to the results of the study. This part is perhaps the most critical aspect of the research in the study.

In this research, the regardless of my field of the study discussed about, the result of the existing active monitoring load balancing algorithm and the modified active monitoring load balancing algorithm with compare and contrast to the other load balancing algorithms.

The first thing that we done is, try to understand the main idea of the load balancing algorithm, especially the active monitoring load balancing algorithm in cloud computing environment. In cloud computing, how the load balancing algorithm works at virtual machine level in cloud computing environment.

In the research to make useful interpretation of data, it is important to select the proper statistical tool. In this study we are using the CloudAnalyst tool to compare and contrast the existing different types of load balancing algorithms and the modified active monitoring load balancing algorithm. And then by configuring the simulation tool to understand the performance of the load balancing algorithms by selecting the service broker policy. From the service broker policy such that: the overall data center processing time, overall response time and others are selected. In order to configure the simulation tool using six (6) different geographical locations in the real world. The rest of the other configuration on the simulation tool is clearly described in (5.2) simulation setup.

5.1 Implementation Plan for the Proposed Algorithm

To achieve the performance of the active monitoring load balancing algorithm and to recommend depend on the final result of the load balancing algorithm, experiment is carried out in the simulator. For the experimentation purpose the CloudAnalyst simulator tool plays a major role. To modify the active monitoring load balancing algorithm we used java language. After modifying the load balancing algorithm, we are adding the new load balancing algorithm to the CloudAnalyst tool. After adding the new load balancing algorithm then we compare and contrast the existing load balancing algorithms and the new one to understand the performance of the algorithms.

In cloud computing environment as the cloud infrastructure distributed, the requests will be coming from all the geographical locations in the world.

In our study experiment to give the real time scenario depends on six different geographical locations in the real world.

The six different geographical locations are: (Africa, Asia, Europe, North American, Oceania and South American). From the hypothetical applications in our experimentation, we consider the twitter users. The twitter is better than other social media.

For instance when we use twitter we don't have to face the problem with bunch of advertisements like that of the other social networking sites. In the CloudAnalyst a single time zone is considered for all user locations [4].

The CloudAnalyst comes with three important menus: configure simulation, define internet characteristics and run simulation. These menus are used for setting of the entire simulation process for the load balancing algorithms in cloud computing environment. The tool provides us the feature of switching algorithms according to our requirements.

According to [29] the basic components of the CloudAnalyst are as follows:

- **Regions:** In the CloudAnalyst simulator tool the whole world split into the six regions. These regions coincide with six main continents in the world. The main entities related to these areas are DC and UB. This categorization plays an important role for maintaining the level of real-based for a very large duration simulation in load balancing algorithms [32].
- **GUI Packages:** For the graphical user interface the GUI package in CloudAnalyst is mainly responsible.
- **User Bases:** When we see user bases in CloudAnalyst simulator the collection of users, considered to be as one individual unit, which are involved in the simulation. Here the simulation considers being in the real time [32]. Thousands of users can be represented as a single user base. But, all these users are configured as a single unit i.e. a single user base. Generally, the user bases are used to be considered as they represent a large number of users in the world.
- **Internet Settings:** In CloudAnalyst simulator internet is the idea for the actual real world. Here when we see internet, it implements all those features which are important to that particular simulation.

The CloudAnalyst also manages to create the replicas of the internet traffic, which seems to be routing around all over the world through hosting appropriate amount of data transmissions. Among six areas the customers can also configure the transmission latency [32].

- **Data Center Controller:** When we see data center controller, it is a collection of servers hosting different application. And also to subscribe different application an end user connects to the datacenter. A datacenter may exist at a large distance from the clients. It generally controls datacenter activities [24].
- **VM Load Balancer:** In datacenter controller the VM load balancer used to determine which virtual machine should be assigned the next cloudlet for processing.
- **Simulation:** The simulation plays an important role in order to create and execute the request. The main objective of simulator is to test the implementation work in the absence of the required environment in load balancing algorithms.
- **Service Broker:** Service Broker used to manage all the traffic that is moving between the data centers and the user bases based on the overall response time and data center controller and others [32].

According to [29] the control panel options in the CloudAnalyst tool are as follows:

- **Configure Simulation:** This is one of the control panel options in the CloudAnalyst tool, which takes us to the configure simulation screen.
- **Define Internet Characteristics:** In CloudAnalyst tool takes us to the internet Characteristics Screen.
- **Run Simulation:** After filling all the necessary data in CloudAnalyst tool, run simulation used to start the test in the CloudAnalyst tool.
- **Exit:** The exit button used to close the simulation screen at all.

As [17] described about the CloudAnalyst, “*it is very flexible and it provides, that of the virtual machines, data centers, band width and many more others for experimentation purpose*”.

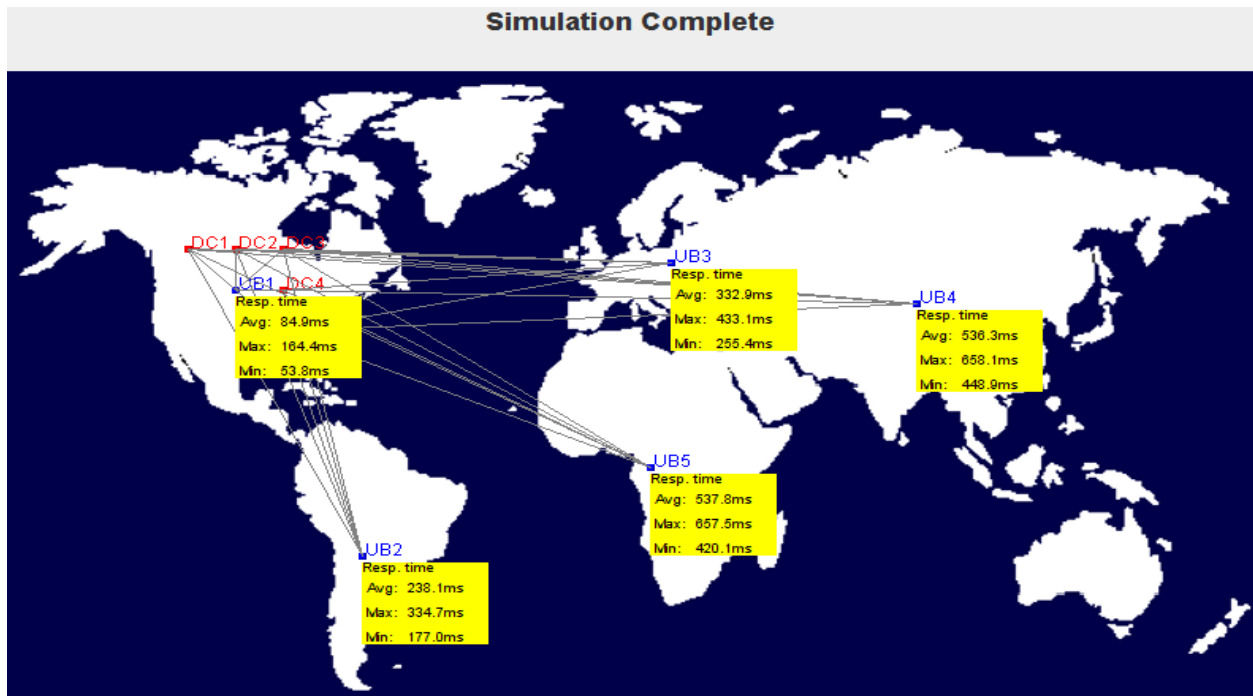


Figure 14: Output Screen after Run Simulation

In our study we also used the cloudsim simulation for different purposes like that of to select the least loaded virtual machine among the available virtual machines. To assign the request to the selected virtual machine and if, all the virtual machine reached in maximum limitation to accept request and to put the tasks/requests on buffer.

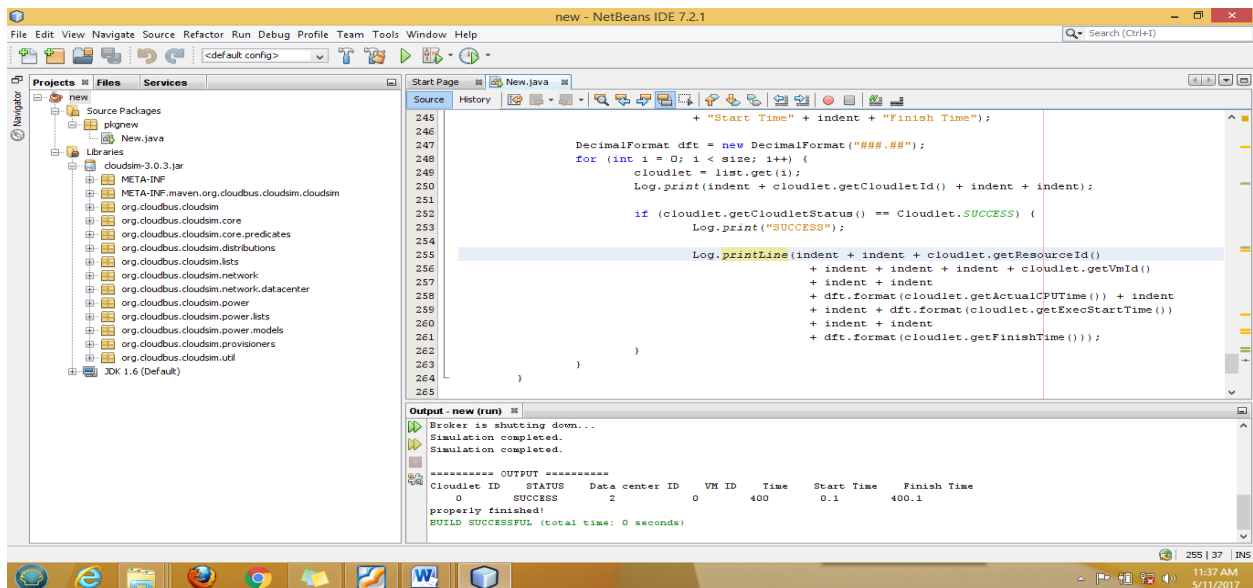


Figure 15: NetBeans IDE for Request Assignment

In our experiment for the load balancing algorithm we are using the CloudAnalyst simulation setup. The results and analysis of simulation setup are analyzed with respect to the efficient utilization of the virtual machines by avoiding the complexity of the active monitoring load balancing algorithm. And then comparing with the existing load balancing algorithm and carried out for a period of 2 (two) hours. And taking different numbers of users, 5 (five) data centers, i.e. data center one (DC1), data center two (DC2), data center three (DC3), data center four (DC4) and data center five (DC5) having 15, 30, 45, 60 and 75 numbers of virtual machines respectively.

And the other parameters are filled and which are fixed according to the next table as shown:

5.2 Simulation Setup

Table 2: Parametres and values in CloudAnalyst simulator

<i>Parameter</i>	<i>Value Passed</i>
VM-image size	10000 Mgb
VM-memory	1024 KiB
VM-bandwidth	1000 Mgbps
Service broker policy	Overall response time Overall data center processing time
Data center architecture	x86
Data center-OS	Linux
Data center-VMM	Xen
Data center- No of VMs	DC1-15 DC2-30 DC3-45 DC4-60 DC5-75
Data center-memory per machine	204800mb
Data center-storage per machine	100000000mb
Data center-available bandwidth per machine	1000000mb
Data center-processor speed	10000 GHz
Data center-VM policy	Time shared
User grouping factor in user bases	1000
Request grouping factor in data centers	500
Executable instruction length per request	500 bytes

5.3 Outputs and Results

After performing the six (6) different experiments using the CloudAnalyst simulator tool using the above parameters successfully, we get the overall response time and the overall data center processing time of the different load balancing algorithms.

To understand the result of the load balancing algorithms we used different cases:

Case one (1): By using same number of processors to find out the performance of the load balancing algorithms. To apply this using the overall response time for active monitoring load balancing algorithm, throttled load balancing algorithm and round robin load balancing algorithm. To do this using 175 (one hundred seventy five) user bases (each user base represents thousands of peoples). The overall response time of the throttled load balancing algorithm is better than that of the other load balancing algorithm based on the existing load balancing algorithms. i.e. dual core processors.

Table 3: Overall response time for case-1

Setup Description	Overall Response Time (in ms)								
	Round Robin			Active Monitoring			Throttled		
Exp1	<i>Avg</i>	<i>Min</i>	<i>Max</i>	<i>Avg</i>	<i>Min</i>	<i>Max</i>	<i>Avg</i>	<i>Min</i>	<i>Max</i>
3 user bases	290.23	60.76	658.46	87.97	60.32	129.01	87.77	60.32	129.01
Exp2	125.08	56.38	270.83	125.06	56.38	270.83	125.05	56.38	270.83
12 user bases									
Exp3	341.60	59.76	646.90	187.30	49.16	401.97	120.31	55.88	273.77
20user bases									
Exp4	339.03	253.72	480.39	178.91	57.82	266.39	124.17	57.38	527.89
60user bases									
Exp5	333.13	235.72	456.60	123.28	55.45	267.15	123.24	55.45	267.15
50user bases									
Exp6	339.77	256.52	477.22	136.42	84.75	130.32	124.89	57.26	258.15
30user bases									

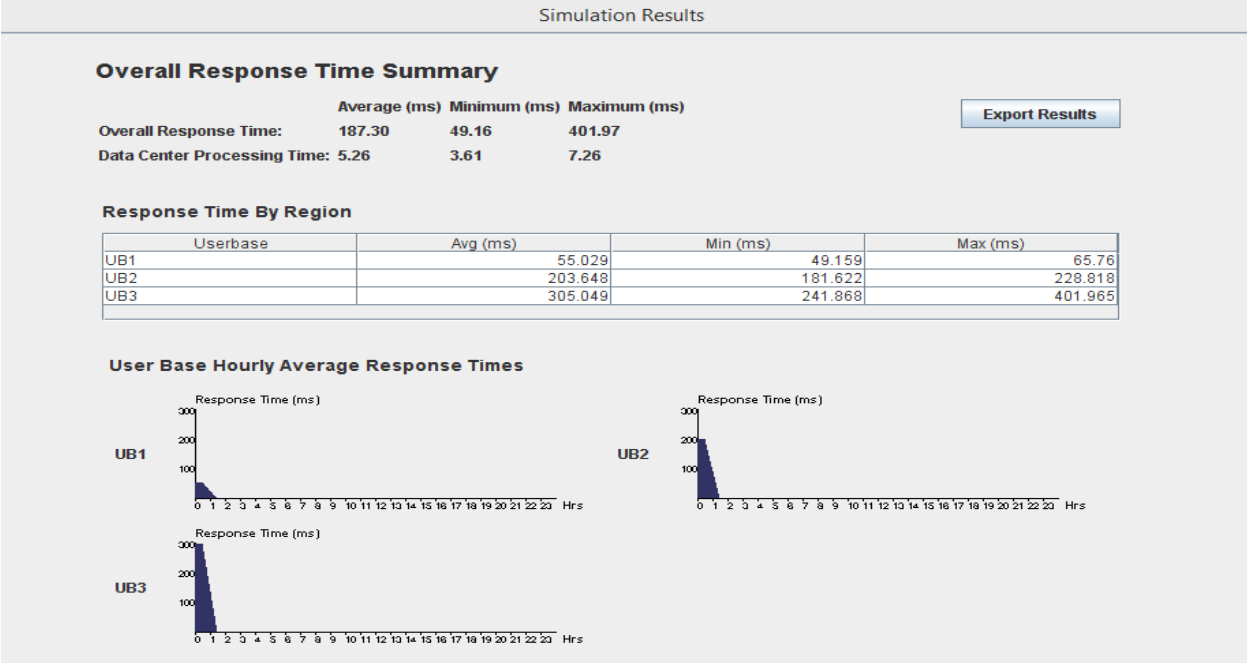


Figure 16: Simulation Result for case one

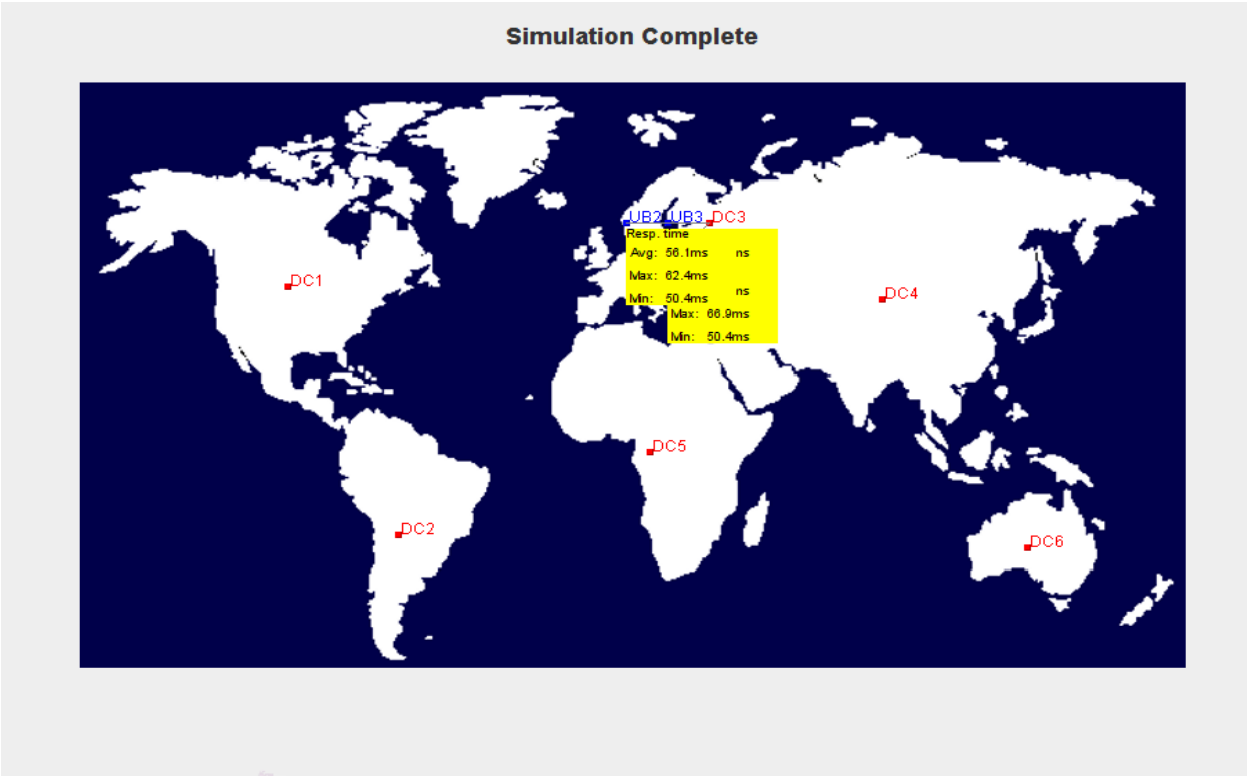


Figure 17: Simulation Result for 3 UBs and 6 DC

Case two (2): Using the overall data center processing time with same number of processors (dual core processors) to find out the performance of the load balancing algorithms.

To do this using the overall data center processing time for active monitoring load balancing algorithm, round robin load balancing algorithm and throttled load balancing algorithm.

With this experiment the overall data center processing time for the throttled load balancing algorithm is better than the active monitoring and round robin load balancing algorithms.

Table 4: Overall data center processing time for case-2

Setup Description	Overall Data Center processing Time(in ms)								
	Round Robin			Active Monitoring			Throttled		
Exp1	<i>Avg</i>	<i>Min</i>	<i>Max</i>	<i>Avg</i>	<i>Min</i>	<i>Max</i>	<i>Avg</i>	<i>Min</i>	<i>Max</i>
3 user bases	39.02	13.51	108.32	38.05	0.00	72.00	38.05	0.00	72.00
Exp2	50.56	13.75	118.39	50.56	13.75	118.39	50.56	13.75	118.39
12 user bases									
Exp3	49.12	13.51	131.26	48.25	13.94	118.15	48.25	13.94	118.15
20user bases									
Exp4	59.84	12.19	120.95	50.90	12.69	121.46	50.86	13.51	122.40
60user bases									
Exp5	53.99	6.41	123.33	49.74	12.94	121.46	49.64	12.94	121.46
50user bases									
Exp6	80.48	13.63	119.08	297.62	46.16	612.02	50.48	13.69	118.40
30user bases									

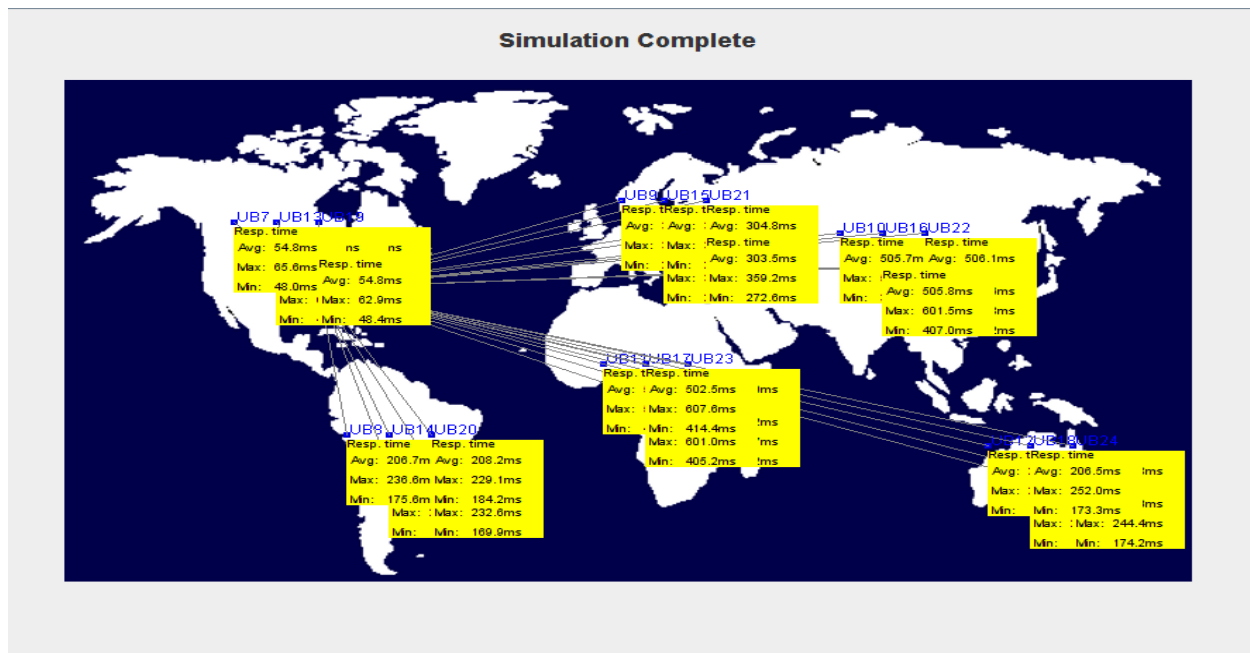


Figure 18: Simulation Result for 30 UBs

Simulation Results				
Overall Response Time Summary				
	Average (ms)	Minimum (ms)	Maximum (ms)	
Overall Response Time:	297.62	46.16	612.02	Export Results
Data Center Processing Time:	5.19	3.61	7.51	
Response Time By Region				
Userbase	Avg (ms)	Min (ms)	Max (ms)	
UB10	505.683	395.824	576.024	
UB11	508.855	449.518	567.218	
UB12	203.548	176.87	232.021	
UB13	55.045	46.159	66.51	
UB14	207.558	169.82	243.371	
UB15	300.858	237.723	347.768	
UB16	512.885	427.57	587.319	
UB17	502.481	414.417	607.57	
UB18	206.453	173.267	251.97	
UB19	55.095	48.411	67.908	
UB1	55.325	48.86	62.959	
UB20	208.207	184.216	229.072	
UB21	304.806	264.519	366.088	
UB22	506.109	437.469	612.016	
UB23	507.979	422.117	588.814	
UB24	204.777	175.918	229.669	
UB25	202.541	169.868	232.621	
UB26	303.454	272.566	359.17	
UB27	510.577	428.225	597.823	
UB28	517.192	433.222	599.669	
UB29	206.321	174.169	244.422	
UB2	203.6	179.32	241.121	
UB30	54.798	48.409	62.911	
UB3	300.95	255.722	361.42	
UB4	505.846	407.017	601.465	
UB5	505.457	405.169	601.039	
UB6	207.099	177.97	242.321	
UB7	54.772	48.01	65.611	
UB8	206.746	175.619	236.567	
UB9	306.954	265.968	360.97	

Figure 19: Overall response time summary for 30 UBs

Case three (3): In this case we consider all the virtual machine having different number of processors. Such that: DC1 having 2 and 4 core processors, DC2 having 2 core processors, DC3 having 2, 4 and 6 core processors, DC4 having 2 and 6 core processors and finally the DC5 having 6 processors for the active monitoring, round robin and throttled load balancing algorithms.

Based on overall response time experiment for the algorithms the throttled load balancing algorithm is better than the active monitoring load balancing algorithm. And the active monitoring load balancing algorithm is better than round robin load balancing algorithm.

Table 5: Overall response time for case-3

Setup Description	Overall Response Time (in ms)								
	Round Robin			Active Monitoring			Throttled		
Exp1 3 user bases	Avg	Min	Max	Avg	Min	Max	Avg	Min	Max
	87.56	52.36	227.27	87.44	52.36	217.72	86.97	52.36	217.72
Exp2 12 user bases	118.59	48.51	262.52	110.99	48.51	262.52	109.57	48.51	262.52
Exp3 20user bases	115.85	50.16	268.94	109.37	49.06	265.02	108.61	49.06	265.02
Exp4 60user bases	117.75	49.26	257.82	109.60	49.26	257.82	109.57	49.26	257.82
Exp5 50user bases	117.46	48.71	265.87	109.72	48.01	258.22	108.56	48.01	258.22
Exp6 30user bases	118.00	49.91	256.68	110.14	48.16	253.52	109.82	48.16	253.52

Case four (4): In this case also we consider all the virtual machines having different number of processors based on overall data center processing time experiment for the active monitoring load balancing algorithm, throttled load balancing and round robin load balancing algorithm.

Such that: DC1 having the mixture of 2 core and 4 core processors, whereas DC2 having only 2 core processors, DC3 have 2, 4 and 6 core processors, DC4 having 2 and 6 core processors and finally DC5 having 6 core processors.

From this test the result shows that the throttled load balancing algorithm is better for the overall data center processing time than active monitoring and round robin load balancing algorithms.

Table 6: Overall data center processing time for case-4

Setup Description	Overall Data Center processing Time(in ms)								
	Round Robin			Active Monitoring			Throttled		
Exp1 3 user bases	Avg	Min	Max	Avg	Min	Max	Avg	Min	Max
	37.36	0.00	95.77	34.89	0.00	95.77	33.92	0.00	95.77
Exp2 12 user bases	42.51	7.80	98.51	34.70	7.45	95.26	33.63	7.45	95.26
Exp3 20user bases	42.64	7.90	103.52	35.19	7.55	103.52	34.53	7.55	103.52
Exp4 60user bases	42.16	7.21	103.52	33.70	6.50	103.52	33.51	6.50	103.52
Exp5 50user bases	42.33	7.61	103.27	34.14	6.50	103.27	33.91	6.50	103.27
Exp6 30user bases	42.40	7.20	103.27	34.04	7.20	103.27	33.62	7.20	103.27

As we understand from the above experiment based on the interpretation and analysis of the existing active monitoring load balancing algorithm performance is not good. This is because at the time of number of requests/tasks increases on a virtual machines.

So to modify and increase the performance of the active monitoring load balancing algorithm we are using the following steps.

Case five (5): We try to solve the problem by using buffer in active monitoring load balancing algorithm process. At the time of number of request comes from the users and then if, the virtual machine reached in maximum limitation to accept any request, put the request to buffer until VM notifies ok to accept request.

Case six (6): Like that of the existing load balancing algorithm experiment by using same number of processors. To find out the performance of the load balancing algorithms using the overall response time, for modified active monitoring load balancing algorithm, throttled load balancing algorithm and round robin load balancing algorithm. To do this using 175 user bases (each user base represents thousands of people).

The overall response time of the modified active monitoring load balancing algorithm and the throttled load balancing algorithms are better than the other existing load balancing algorithms.

Table 7: Overall response time for case-6

Setup Description	Overall Response Time (in ms)								
	Round Robin			Modified Active Monitoring			Throttled		
Exp1 3 user bases	<i>Avg</i>	<i>Min</i>	<i>Max</i>	<i>Avg</i>	<i>Min</i>	<i>Max</i>	<i>Avg</i>	<i>Min</i>	<i>Max</i>
	290.23	60.76	658.46	87.77	60.32	129.01	87.77	60.32	129.01
Exp2 12 user bases	125.08	56.38	270.83	125.09	56.38	270.83	125.05	56.38	270.83
Exp3 20user bases	341.60	59.76	646.90	120.32	55.88	273.77	120.31	55.88	273.77
Exp4 60user bases	339.03	253.72	480.39	178.17	57.82	266.39	178.17	57.38	527.89
Exp5 50user bases	333.13	235.72	456.60	123.30	55.45	267.15	123.29	55.45	267.15
Exp6 30user bases	339.77	256.52	477.22	124.88	84.75	130.32	124.89	57.26	258.15

Case seven (7) : Using the overall data center processing time with same number of processors for the round robin load balancing algorithm, the modified active monitoring load balancing algorithm and throttled load balancing algorithm. With this experiment the overall data center processing time for the modified active monitoring load balancing algorithm is better than that of the other load balancing algorithms.

Table 8: Overall data center processing time for case 7

Setup Description	Overall Data Center processing Time(in ms)								
	Round Robin			Modified Active Monitoring			Throttled		
Exp1 3 user bases	Avg	Min	Max	Avg	Min	Max	Avg	Min	Max
	39.02	13.51	108.32	38.05	0.00	72.00	38.05	0.00	72.00
Exp2 12 user bases	50.56	13.75	118.39	50.46	13.75	118.39	50.56	13.75	118.39
Exp3 20user bases	40.12	13.51	131.26	48.25	13.94	118.15	48.25	13.94	118.15
Exp4 60user bases	39.84	12.19	120.95	50.50	12.69	121.46	50.86	13.51	122.40
Exp5 50user bases	33.99	6.41	123.33	49.64	12.94	121.46	49.64	12.94	121.46
Exp6 30user bases	40.48	13.63	119.08	50.46	0.00	79.01	50.48	13.69	118.40

Case eight (8): In this case we consider all the virtual machines having different number of processors. Such that: DC1 having 2 and 4 core processors, DC2 having 2 core processors, DC3 having 2, 4 and 6 core processors, DC4 having 2 and 6 core processors and finally the DC5 6 processors for the modified active monitoring load balancing algorithm, round robin load balancing algorithm and throttled load balancing algorithm.

Based on overall response time experiment for the algorithms the throttled load balancing algorithm and the modified active monitoring load balancing algorithms are better and similar than that of the round robin load balancing algorithm.

Table 9: Overall response time for case 8

Setup Description	Overall Response Time (in ms)								
	Round Robin			Modified Active Monitoring			Throttled		
Exp1	Avg	Min	Max	Avg	Min	Max	Avg	Min	Max
3 user bases	87.56	52.36	227.27	86.96	52.36	217.72	86.97	52.36	217.72
Exp2	118.59	48.51	262.52	109.69	48.51	262.52	109.57	48.51	262.52
12 user bases									
Exp3	115.85	50.16	268.94	109.00	49.06	265.02	108.61	49.06	265.02
20user bases									
Exp4	117.75	49.26	257.82	109.60	49.26	257.82	109.57	49.26	257.82
60user bases									
Exp5	117.46	48.71	265.87	109.72	48.01	258.22	109.56	48.01	258.22
50user bases									
Exp6	118.00	49.91	256.68	109.89	48.16	253.52	109.82	48.16	253.52
30user bases									

Case nine (9): The overall data center processing time for the all virtual machines having different number of processors for different load balancing algorithms also analyzed. From this test the result shows that the modified active monitoring load balancing algorithm is better than the other load balancing algorithms for the overall data center processing time in cloud computing environment.

Table 10: Overall data center processing time for case 9

Setup Description	Overall Data Center processing Time(in ms)								
	Round Robin			Modified Active Monitoring			Throttled		
Exp1	Avg	Min	Max	Avg	Min	Max	Avg	Min	Max
3 user bases	37.36	0.00	95.77	33.89	0.00	95.77	33.92	0.00	95.77
Exp2	42.51	7.80	98.51	33.60	7.45	95.26	33.63	7.45	95.26
12 user bases									
Exp3	42.64	7.90	103.52	35.00	7.55	103.52	34.53	7.55	103.52
20user bases									
Exp4	42.16	7.21	103.52	33.41	6.50	103.52	33.51	6.50	103.52
60user bases									
Exp5	42.33	7.61	103.27	33.14	6.50	103.27	33.91	6.50	103.27
50user bases									
Exp6	42.40	7.20	103.27	34.04	7.20	103.27	33.62	7.20	103.27
30user bases									

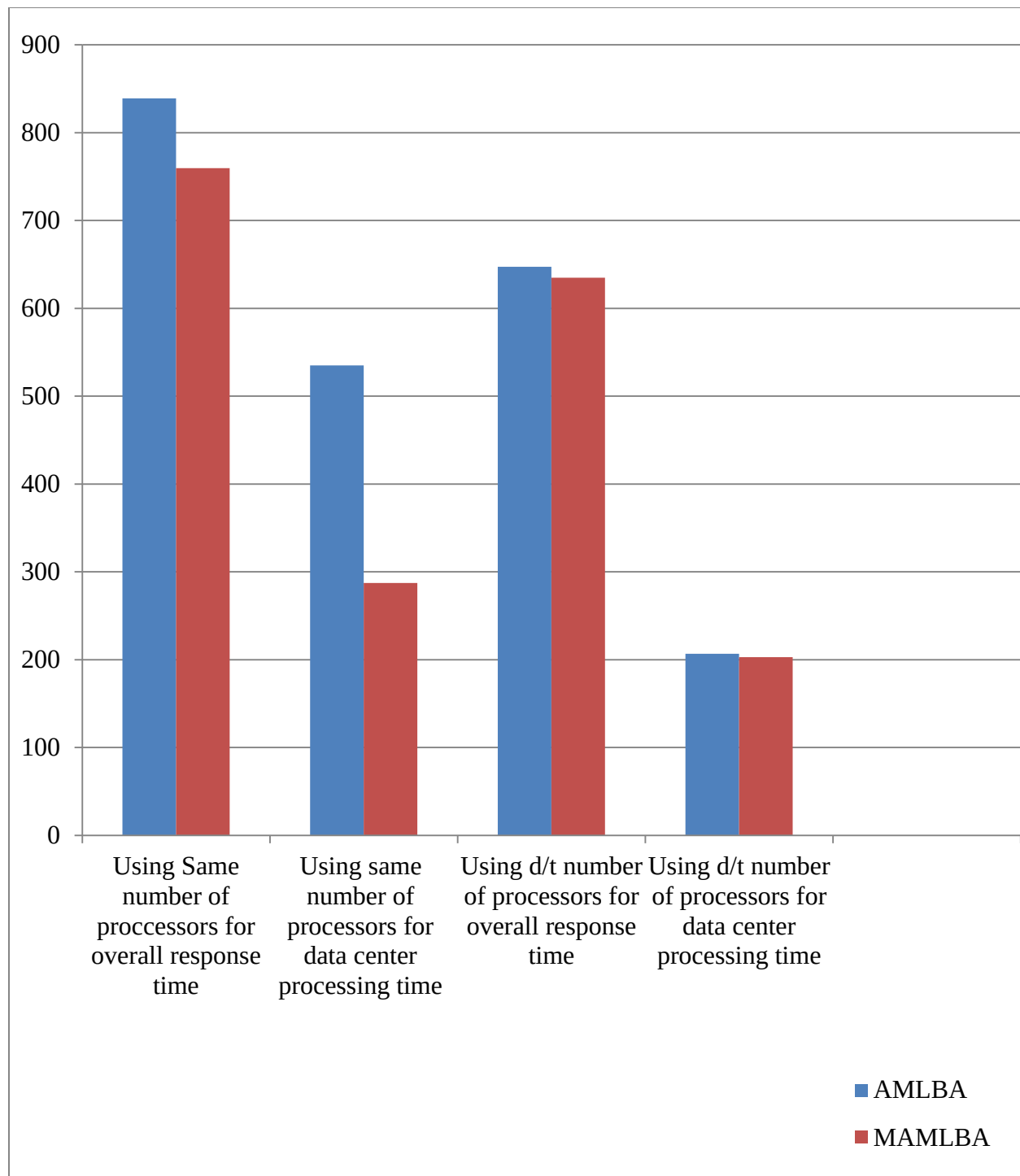


Figure 20: Final test result

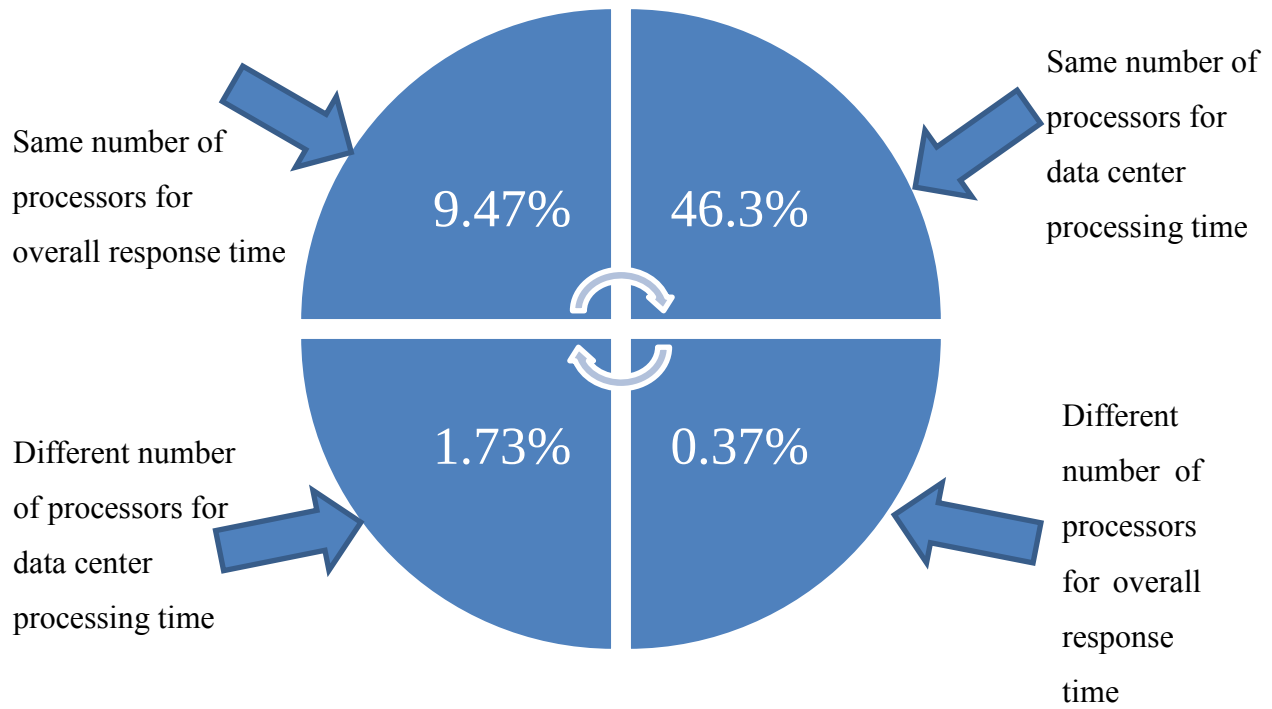


Figure 21: Final test output in percent

5.4 Discussion and Interpretation

Based on the interpretation and analysis of the experiment on existing load balancing algorithm as we discussed in table 3,4,5 and 6 the active monitoring load balancing algorithm performance is not good. This is because when number of cloud users increases to access cloud resources on a virtual machine the complexity of the algorithms also increased.

On the other hand on the modified active monitoring load balancing algorithm at the time of number of requests/tasks increases on virtual machines and the virtual machines reached in maximum limitation to accept any request the algorithm uses buffer.

In table 7 and 9 the overall response time of the modified active monitoring and the throttled load balancing algorithms are better than the other existing load balancing algorithms. In the same way in table 8 and 10 when we see the performance of the existing load balancing algorithms and the modified active monitoring load balancing algorithm, the overall data center processing time for the modified active monitoring load balancing algorithm is better than that of the existing load balancing algorithms.

In general, for the above discussion by testing forty eight (48) experiments using one thousand four hundreds (1400) user bases (each user base represents thousands of peoples) to understand the performance of the load balancing algorithms in cloud computing environment.

To do so using same number of processors and different number of processors depending on among the service broker policy (which is data center processing time and overall response time). Here using same number of processors and different number of processors for data center processing time and overall response time of the modified active monitoring load balancing algorithm is better than the existing active monitoring load balancing algorithm.

CHAPTER SIX

6. CONCLUSION AND RECOMMENDATION

6.1 Conclusion

Cloud is subject to user requirement, load balance and other constraints that have direct effect on user consumption of resources controlled by cloud provider [11, 50].

As [6] described about cloud computing, it is a modern style of using the power of internet and wide area network (WAN) to access resources remotely. And also it is a new solution and strategy to achieve high availability, flexibility, cost reduced and on demand scalability. Cloud computing has become very popular in recent years as it offers greater flexibility and availability of computing resources. It is also a general term used to describe a new class of network based computing that takes place over the internet [4].

In cloud computing environment balancing load is very important. Now a days, cloud computing is one of the most very popular and rapid growing technology. From all over the world clients request for various cloud computing services. In present situation the load balancing algorithms are built to allocate the user requests/tasks to cloud resources in very efficient and intelligent way [3].

According to [8] explanation about cloud computing, it is also intended to enable the computing across largely and diverse resources and dynamic stability. Load balancing is one of the main issues related to cloud computing. The load can be a memory, CPU capacity, network or delay load [4].

In this research, we begun by defining the specific research problem and justify the value of a solution analysis of literatures on a load balancing algorithm adoption in cloud computing environment. And in cloud computing environment, analyzed the current status of the load balancing algorithms. To do this there are many limitations in delivering to describe the utilization concept of the load balancing algorithm and to modify the active monitoring load balancing algorithm.

Those numbers of limitations are raised because of number of reasons. Such as: lack of experience of individual research project, lack of full text paper, shortage of on job (regular) training, time given for the study and others.

Since, based on the problem definition, we atomize the problem conceptually to capture its complexity. For developing the proposed load balancing algorithm in cloud computing environment that can effectively provide a solution. And then define the objective for a solution to develop the proposed load balancing algorithm in cloud computing environment for the knowledge of the state of problems and the current solutions. To observe and measure how well the algorithm supports a solution to the problem is very important. This activity involves comparing the objectives of a solution to actual observed results from the use of cloud computing resources in the demonstration.

There are different types of load balancing algorithms are analyzed based on the CloudAnalyst simulator tool and to get results. To analyzed the efficient utilization of the virtual machines by avoiding the complexity of the active monitoring load balancing algorithm at the time of number of requests/tasks increase on a virtual machine. To analyze the least loaded virtual machine among the available virtual machines and to allocate users request to this selected virtual machine the cloudsim simulator is used. If the virtual machine reached in maximum limitation to accept any request the cloudsim simulation used to assign request to buffer.

The study took the qualitative evaluation analysis techniques. Which include: a comparison of the existing load balancing algorithm with the proposed load balancing algorithm functionality with the objectives solution. Qualitative performance measures, such as: extents of overall response time and the overall data center processing time of the load balancing algorithms in cloud computing. Conceptually, such kind of evaluation could include any of the appropriate logical proof.

As a general, if the number of requests/tasks increases on a virtual machine the existing active monitoring load balancing algorithm is more complex. By introducing the modification and utilization concept of this algorithm we would like to present a simple efficient active monitoring load balancing algorithm in cloud computing environment.

As a final, this research study plays an important role in order to investigate the benefits for us that can we use cloud computing resources. To get the resources we are using the load balancing algorithms at virtual machine level in cloud environment in each our day to day life. And also it gave us the awareness of the cloud computing environment and the load balancing algorithms.

6.2 Recommendations

However the results of this study is good and better, further work needs to be done. To realize the proposed algorithm for the variants of the active monitoring load balancing algorithm at virtual machine level in cloud environment. And propose a fair active monitoring load balancing algorithm is needed.

In active monitoring load balancing algorithm there are variants of load balancing algorithms available introducing different techniques focusing on different measuring parameters of it. This entire algorithm follows different strategies to improve the active monitoring algorithm's performance. So for these variants of active monitoring load balancing algorithm, I recommend that to prepare fair active monitoring load balancing algorithm for better implementation.

6.3 Future Work

In this study the proposed algorithm can still be improved by taking some other more dynamic situations of the incoming requests in active monitoring load balancing algorithm. And how the algorithm responses, if we mix both static and dynamic loads. Instead of CPU utilization some other different parameters can be considered for a virtual machine allocation to obtain the utilization. In future, we need to develop a load balancing technique, which is an advantage at the cloud service provider in terms of the cost. Using a load balancing algorithm to improve the load distribution performance in cloud computing environment depend on the server overflow problem. As our future work, we are planning to improve the load balancing algorithm to make it more suitable for cloud environment and more efficient in terms of process migration.

References

1. Dharmesh Kashyap, Jaydeep Viradiya, (2014). "A Survey of Various Load Balancing Algorithms In Cloud Computing", International Journal of Scientific & Technology Research Volume 3.
2. Soumya Ranjan Jena Asst. Professor M.I.E.T, (2013). "Response Time Minimization of Different Load Balancing Algorithms in Cloud Computing Environment", Computer Applications (0975 – 8887) Volume 69– No.17.
3. Tanveer Ahmed, Yogendra, (2012). "Analytic Study of Load Balancing Techniques Using Tool Cloud Analyst", Singh/International Journal of Engineering Research and Applications (IJERA), , Vol. 2, Issue 2, Available at www.ijera.com.
4. Akshay Daryapurkar, Mrs. V.M. Deshmukh, (2013). "Efficient Load Balancing Algorithm in Cloud Environment", International Journal Of Computer Science And Applications Vol. 6, No.2.
5. Hafiz Jabr Younis, (2015). Annual report year ended Feb, 2015. "Efficient Load Balancing Algorithm in Cloud Computing", Gaza Deanery of Post Graduate Studies Faculty of Information Technology.
6. Mintu M. Ladani,Vinit Kumar Gupta, (2013). "A Framework for Performance Analysis of Computing Clouds", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: (2278-3075), Volume-2, Issue-6.
7. Divya Chaudhary, Rajender Singh Chhillar, (2013). "A New Load Balancing Technique for Virtual Machine Cloud Computing Environment", Department of Computer Science and Applications, M.D. University, Rohtak, Haryana, India, International Journal of Computer Applications (0975 – 8887) Volume 69– No.23.
8. Bhaskar. R, Deepu. S.R and Dr. B.S. Shylaja, (2012). "Dynamic Allocation Method For Efficient Load Balancing In Virtual Machines For Cloud Computing Environment", Advanced Computing: An International Journal, Vol.3, No.5.
9. MR.Amit A. Kariyani (2013). "Allocation Of Virtual Machines In Cloud Computing Using Load Balancing Algorithm", Information Technology Dharmsinh Desai University Nadiad, India IRACST - International Journal of Computer Science and Information Technology & Security (IJCSITS), ISSN: (2249-9555) Vol. 3, No.1.

10. Jaspreet Kaur et al. (2012). "Comparison of load balancing algorithms in a Cloud", International Journal of Engineering Research and Applications (IJERA).
11. Doddini Probhuling L. (2013). "Load Balancing Algorithms In Cloud Computing", International Journal of Advanced Computer and Mathematical Sciences ISSN (2230-9624) Vol4.
12. Amandeep, Van dana Yadav, Faz Mohammad, (2014). "Different Strategies for Load Balancing in Cloud Computing Environment: a critical Study", International Journal of Scientific Research Engineering & Technology, (IJSRET), ISSN 2278 – 0882 Volume 3 Issue1.
13. Reena Panwar, (2015). "A Comparative Study of Load Balancing Algorithms in Cloud Computing", International Journal of Computer Applications (0975 – 8887) Volume 117 – No. 24.
14. Meenakshi Sharma, Pankaj Sharma, Dr. Sandeep Sharma, (March2012). "Efficient Load Balancing Algorithm in VM Cloud Environment", Dept. of CSE Sri Sai College of Engineering & Technology, Badhani, Pathankot, Punjab, India, IJCST Vol. 3, ISSue-1.
15. Faouzia Benabbou, (2015). "Load Balancing for Improved Quality of Service in the Cloud", (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 6, No. 7.
16. Mr. R.Gnanasundaram, Prof. S.Suresh (2015). "Optimal load balancing in cloud computing by efficient utilization of virtual machines", International Journal of Advanced Technology in Engineering and Science, Volume No.03, Special Issue No. 02.
17. Hemant S. Mahalle, Parag R. Kaveri and Vinay Chavan, (2013). "Load Balancing on Cloud Data Centres", Advanced Research in Computer Science and Software Engineering, Volume3.
18. Minakshi Berwa, Dr. Chander Kant Research Scholar, (2015). "Load Balancing in Cloud Computing", Department of Computer Science and Application, K.U., Kurukshetra, India, volume 6 number 2.
19. Sandip Patel U & P U. (2015). "CloudAnalyst A Survey of Load Balancing Policies", Patel Department of Computer Engineering, Charusat University Changa, Gujarat, India International Journal of Computer Applications (0975 – 8887) Volume 117 – No. 21.

20. Amal Zaouch, (2015). "Load Balancing for Improved Quality of Service in the Cloud", International Journal of Advanced Computer Science and Applications, Vol. 6, No. 7.
21. Ahmad Faraahi, Zeinab Goudarzi, (2014). "Effective load balancing in cloud computing", International Journal of Intelligent Information Systems.
22. Nusrat Pasha, Dr. Amit Agarwal, (2014). "Round Robin Approach for VM Load Balancing Algorithm in Cloud Computing Environment", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 5.
23. Rodrigo N. Calheiros, Rajiv Ranjan, César A. F. De Rose, Rajkumar Buyya, (2009). "CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services".
24. Dr. Ravi Rastogi, Nusrat Pasha, Dr. Amit Agarwal, (May 2014). "Round Robin Approach for VM Load Balancing Algorithm in Cloud Computing Environment", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 5.
25. Atul Raj, Abhay Kumar Agarwal, (2015). "Load Balancing Algorithm in Cloud Computing", International Journal of Computer Applications, (0975 – 8887) Volume 132 – No.2.
26. Bhathiya Wickremasinghe, (2009). "CloudAnalyst: A CloudSim-based Tool for Modelling and Analysis of Large Scale Cloud Computing Environments", Distributed Computing Project, Csse Dept., University of Melbourne.
27. B Preethi, Professor C. Kamalanathan, Dr. S.M Ramesh, S Shanmathi, P Sathiya Bama, (2014). "Optimization of Resources in Cloud Computing Using Effective Load Balancing Algorithms", International Advanced Research Journal in Science, Engineering and Technology Vol. 1, Issue 1.
28. Bhawna Mallick, Ph.D Professor, (2015). "A Comparative Study of Load Balancing Algorithms in Cloud Computing", International Journal of Computer Applications (0975 – 8887) Volume 117 – No. 24.
29. Radha, Dr. Vivek Kumar, (2016). "Efficient VM Load Balancing Algorithm for Dynamic Allocation of Resources in Cloud Computing Environment", International Journal of Innovative Research in Computer and Communication Engineering (An ISO 3297: 2007 Certified Organization) Vol. 4, Issue 6.

30. Jasmin James, Dr.BhupendraVerma, (2012). “Efficient VM Load Balancing Algorithm for A Cloud Computing Environment”, International Journal on Computer Science and Engineering (IJCSE).
31. Shridhar G. Domanal and G. Ram Mohana Reddy, (2013). "Load Balancing in Cloud Computing Using Modified Throttled Algorithm", IEEE, International conference.
32. David R. Karger MIT, (2014). “Simple Efficient Load Balancing Algorithms for Peer-to-Peer Systems”.
33. Gopal Dhaker, Dr. Savita Shiwani, (2014). “Auto-Scaling Load Balancing and Monitoring As service in public cloud”, Computer Science& Engineering Gyan Vihar University, Jaipur 302025, India, Journal of Computer Engineering (IOSR-JCE) e-ISSN: (2278-0661), p- ISSN: 2278-8727 Volume 16, Issue 4.
34. Harsha M B, Dr. Sarojadevi H, (2016). “Previous Allocation Based Active Monitoring Algorithm for Load Balancing in Cloud Computing”, International Journal of Innovative Research in Science, Engineering and Technology (An ISO 3297: 2007 Certified Organization), Vol. 5, Special Issue 9.
35. Abhay Kumar Agarwal, (2015). “A New Static Load Balancing Algorithm in Cloud Computing”, International Journal of Computer Applications, (0975 – 8887) Volume 132 – No.2.
36. Subasish Mohapatra, (May 2013). “Analysis of Different Variants in Round Robin Algorithms for Load Balancing in Cloud Computing”, Dept. Of Cse, Nit, Rourkela, International Journal of Computer Applications (0975 – 8887) Volume 69– No.22.
37. Alemseged Kassahun, (October, 2013). ”Analysis And Modeling For Adoption Of Hybrid Cloud Computing In Educational Institutions”.
38. Alemayehu Abere Mulu, (2014). ”A Cloud Computing Framework for Ethiopian Banking Industry”.
39. Nyaoro B. (2012). “A Framework to Guide Companies on Adopting Cloud Computing Technology”, Faculty of Information Technology, Strathmore University, Nairobi, Kenya.
40. Simar Preet Singh, Anju Sharma and Rajesh Kumar, (2016). “Analysis of Load Balancing Algorithms using Cloud Analyst”, International Journal of Grid and Distributed Computing Vol. 9, No. 9 pp.11-24, <http://dx.doi.org/10.14257/ijgdc>.

41. Pakize, S.R., Khademi, S.M., and Gandomi, (2014). "A Comparison of CloudSim, CloudAnalyst and CloudReports Simulator in Cloud Computing", International Journal of Computer Science and Network Solutions, 2: pp. (19-27).
42. Shiva Prakash, Vikas kumar, (2014). "Modified Active Monitoring Load Balancing with Cloud Computing", Department of Computer Science & Engineering. Malaviya University of Technology, Gorakhpur (U.P.), India, IJSRD-International Journal for Scientific Research & Development| Vol. 2, (2321-0613), Issue 09.
43. Vikas Kumar and Shiva Prakash, (2014). "A Load Balancing Based Cloud Computing Techniques and Challenges", Department of Computer Science & Engineering Madan Mohan Malaviya University of Technology, Gorakhpur, International Journal of scientific research and management (IJSRM) Volume2 Issue5 Pages (815 -824).
44. Rajesh Kumar, Anju Sharma, Simar Preet Singh, (2016). "Analysis of Load Balancing Algorithms using Cloud Analyst", International Journal of Grid and Distributed Computing Vol. 9, No. 9, pp.11-24.
45. Nusrat Pasha, Dr. Amit Agarwal, (2014). "Round Robin Approach for VM Load Balancing Algorithm in Cloud Computing Environment", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 5.
46. Stuti Dave, Prashant Mehta, (2013). "Role of Virtual Machine Live Migration in Cloud Load Balancing", IOSR Journal of Computer Engineering (IOSR-JCE), e-ISSN: (2278-0661), p-ISSN: 2278-8727Volume X, Issue X.
47. Rajender Singh Chhillar, (May 2013). "A New Load Balancing Technique for Virtual Machine Cloud Computing Environment", International Journal of Computer Applications (0975 – 8887) Volume 69– No.23.
48. Rajwinder Kaur and Pawan Luthra, (2014). "Load balancing in Cloud Computing", SBS State Technical Campus/M.tech ,CSE,Student, Ferozepur, India, Proc. of Int. Conf. on Recent Trends in Information, Telecommunication and Computing, ITC.
49. Karthika M.T., Neethu Kurian, Mariya Seby, (June, 2013). "Comparison of Load Balancing and Scheduling Algorithms in Cloud Environment", International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: (2278-3075), Volume-3, Issue-1.
50. Jaakko Jaatmaa, (2010). "Financial Aspects of Cloud Computing Business Models", AaltoUniversity, Information Systems Science.

51. R. Buyya, R. Ranjan, and R. N. Calheiros, (2009). "Modeling and Simulation of Scalable Cloud Computing Environments and the CloudSim Toolkit: Challenges and Opportunities", Proc. of the 7th High Performance Computing and Simulation Conference (HPCS' 09), IEEE Computer Society.
52. Shahapure, N. H., & Jayarekha, P. (2014). "Load Balancing in Cloud Computing".
53. Stuti Dave B H Gardi– India, (May 2014). "Utilizing Round Robin Concept for Load Balancing Algorithm at Virtual Machine Level in Cloud Environment", College of Engineering & Technology, International Journal of Computer Applications (0975 – 8887) Volume 94 – No 4.

Appendixes

Appendix 1: Sample code of the Modified AMLBA

*/** The code that holds the count current modified active monitoring load balancing algorithm allocations on each virtual machine.*/*

Public class ModifiedActiveVmLB extends VmLoadBalancer implements
CloudSimEventListener {

private Map<Integer, Integer>currentAllocationCounts;

private Map<Integer, VirtualMachineState > vmStatesList;

public ActiveVmLoadBalancer(DatacenterController dcb){

dcb.addCloudSimEventListener(this);

this.vmStatesList = dcb.getVmStatesList();

this.currentAllocationCounts = Collections.synchronizedMap (new HashMap<Integer,
Integer>());

}

*/** If the number of request increases on a virtual machine and all the available virtual machines reached in maximum limitations to accept any request, put the request on buffer.*/*

if (currentAllocationCounts.size() > vmStatesList.size()) {

for (int availableVmId : vmStatesList.keySet()){

if (!currentAllocationCounts.containsKey(availableVmId)){

vmId = availableVmId;

break;

}

}

```

    } else {

        int currCount;

        int minCount = Integer.MAX_VALUE;

        for (int thisVmId : currentAllocationCounts.keySet()){

            currCount = currentAllocationCounts.get(thisVmId);

            if (currCount < minCount){

                minCount = currCount;

                vmId = thisVmId;

            }

        }

    }

    allocatedVm(vmId);

    return vmId;

}

```

@Override

```

public void cloudSimEventFired(CloudSimEvent e) {

    if (e.getId() == CloudSimEvents.EVENT_CLOUDLET_ALLOCATED_TO_VM){

        int vmId = (Integer) e.getParameter(Constants.PARAM_VM_ID);

        Integer currCount = currentAllocationCounts.remove(vmId);

        if (currCount == null){

            currCount = 1;

        }

    }

}

```

```

    } else {

        currCount++;    }

        currentAllocationCounts.put(vmId, currCount); }

else if (e.getId() == CloudSimEvents.EVENT_VM_FINISHED_CLOUDLET){

    int vmId = (Integer) e.getParameter(Constants.PARAM_VM_ID);

    Integer currCount = currentAllocationCounts.remove(vmId);

    if (currCount != null){

        currCount--;

        currentAllocationCounts.put(vmId, currCount);

    }

}

}

}

}

```

Appendix 2: The Integration sample code of the AMLBA to the Tool

```

public interface Constants {

    final String STANDARD_SEPARATOR = “_”;

    final String LOAD_BALANCE_MACTIVE = “Modified Active Monitoring LB”; }

    cmbLoadBalancingPolicy = new JComboBox (new String []{

        nnnnConstants.Load_Balance_MACTIVE,

        loadBalancePolicy.equals(Constants.Load_Balance_MACTIVE) {

        this.loadBalancer = new ModifiedActiveloadBalancer (this); }}

```

```

Constants.Load_Balance_MACTIVE,

loadBalancePolicy.equals(Constants.Load_Balance_MACTIVE) {

this.loadBalancer = new ModifiedActiveloadBalancer (this); }}

```

Appendix 3: Snapshots (Results)

Cloud Analyst

Help

Configure Simulation

Main Configuration | Data Center Configuration | Advanced

Simulation Duration: 2 hours

User bases:

Name	Region	Requests per User per Hr	Data Size per Request (bytes)	Peak Hours Start (GMT)	Peak Hours End (GMT)	Avg Peak Users	Avg Off-Peak Users
UB2	1	60	100	3	9	1000	100
UB3	2	60	100	3	9	1000	100
UB4	3	60	100	3	9	1000	100
UB5	4	60	100	3	9	1000	100
UB6	5	60	100	3	9	1000	100

Add New Remove

Application Deployment Configuration:

Service Broker Policy: Optimise Response Time

Data Center	# VMs	Image Size	Memory	BW
DC1	15	10000	1024	1000
DC2	30	10000	1024	1000
DC3	45	10000	1024	1000
DC4	60	10000	1024	1000
DC5	75	10000	1024	1000

Add New Remove

Cancel Load Configuration Save Configuration Done

Fig1: Main Configuration Simulation using UB.

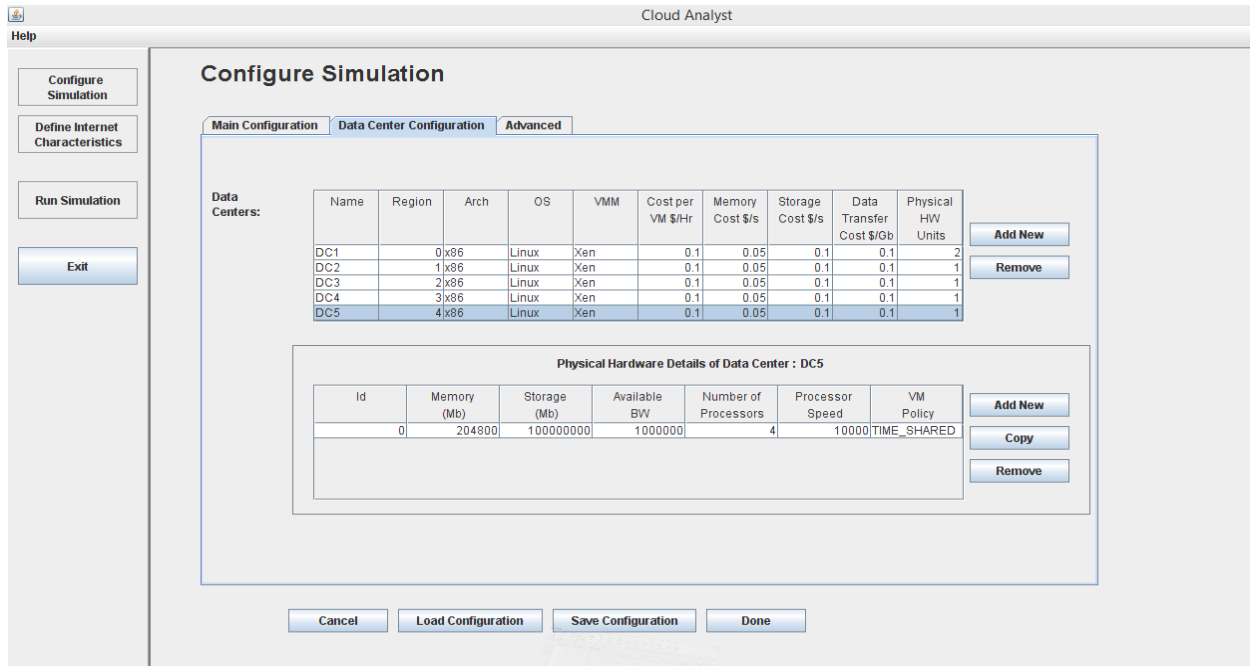


Fig 2: Configuration Simulation using Data Center

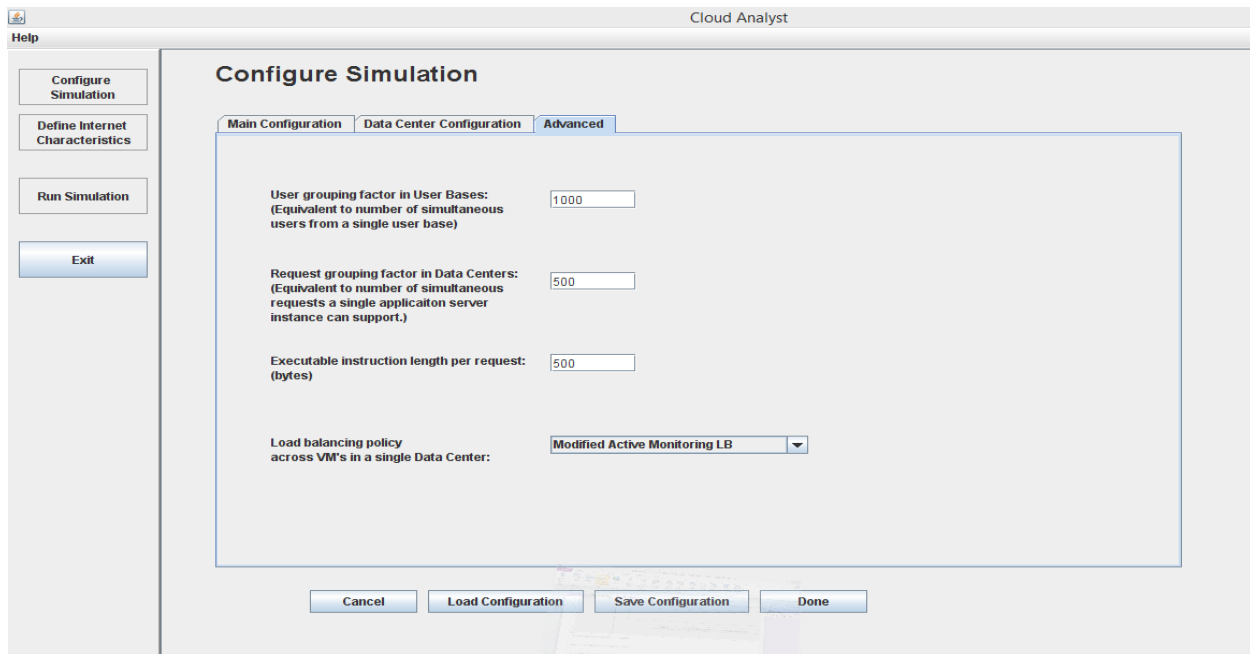


Fig 3: Configuration Simulation using MAMLBA

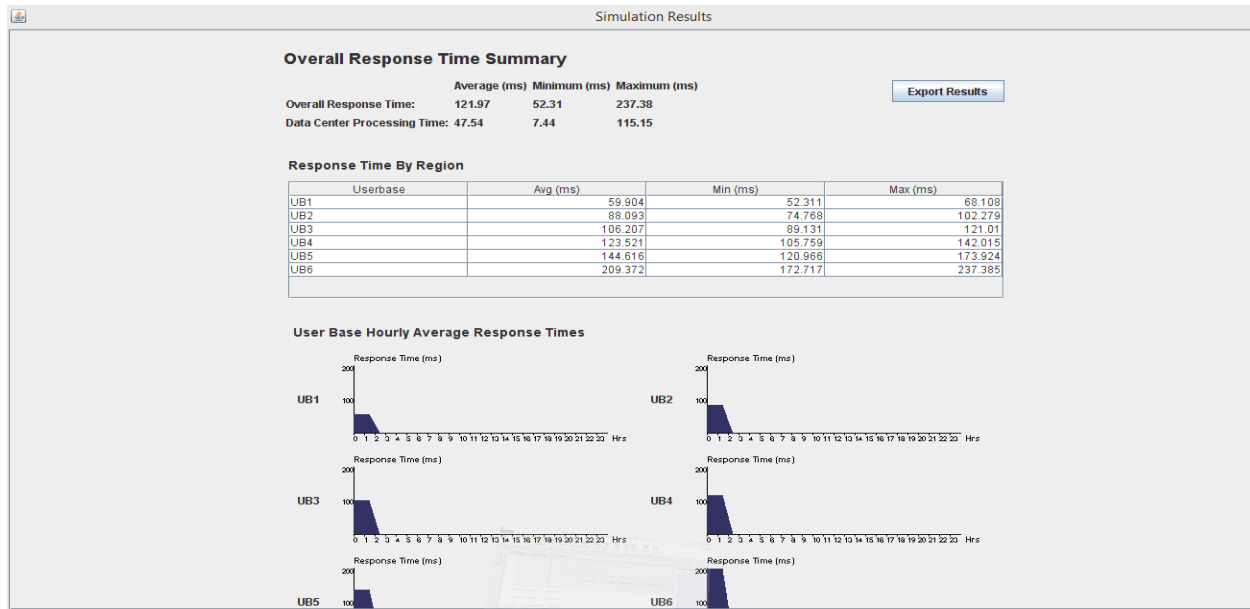


Fig 4: Overall Response Time Summary for Modified Algorithm

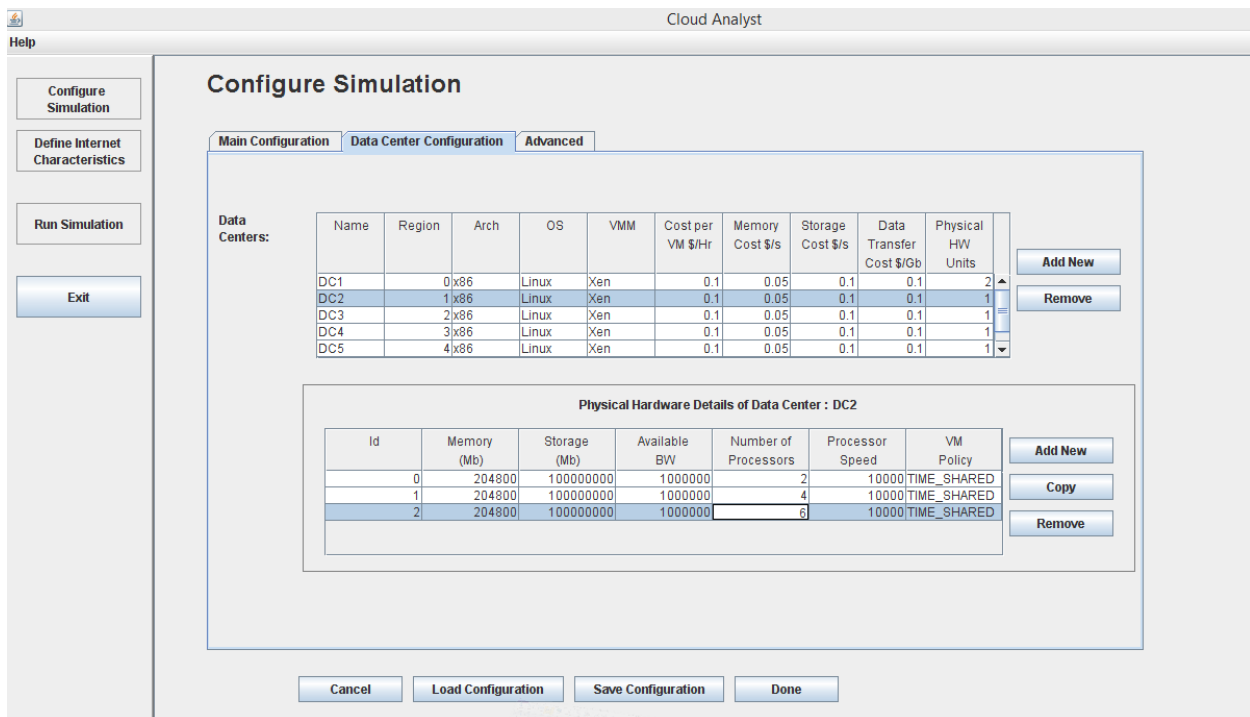


Fig 5: Modified Algorithm with different number of processors

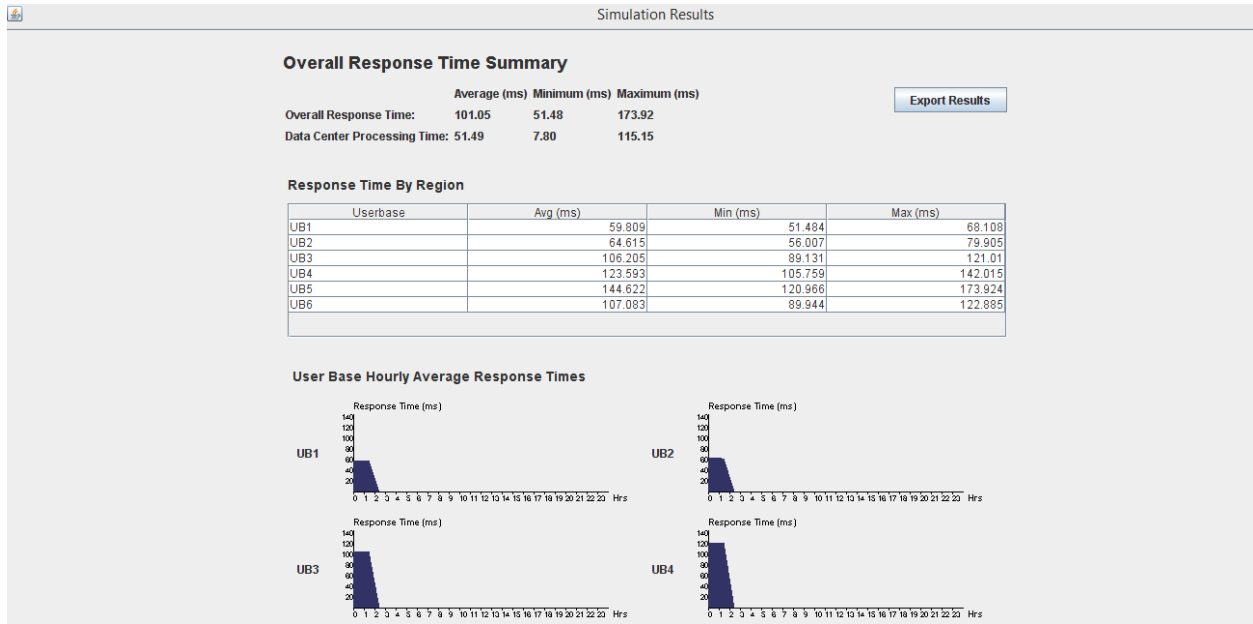


Fig 6: Overall Response Time Summary for Modified algorithm

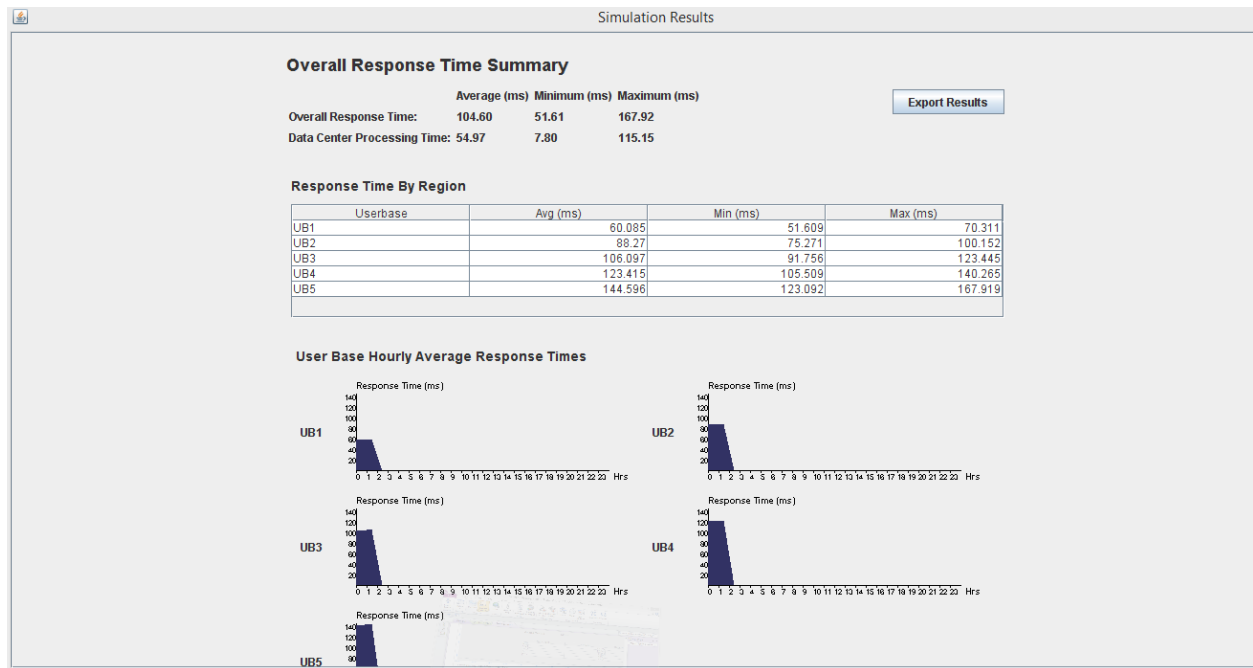


Fig 7: Overall Response Time Summary for existing algorithms

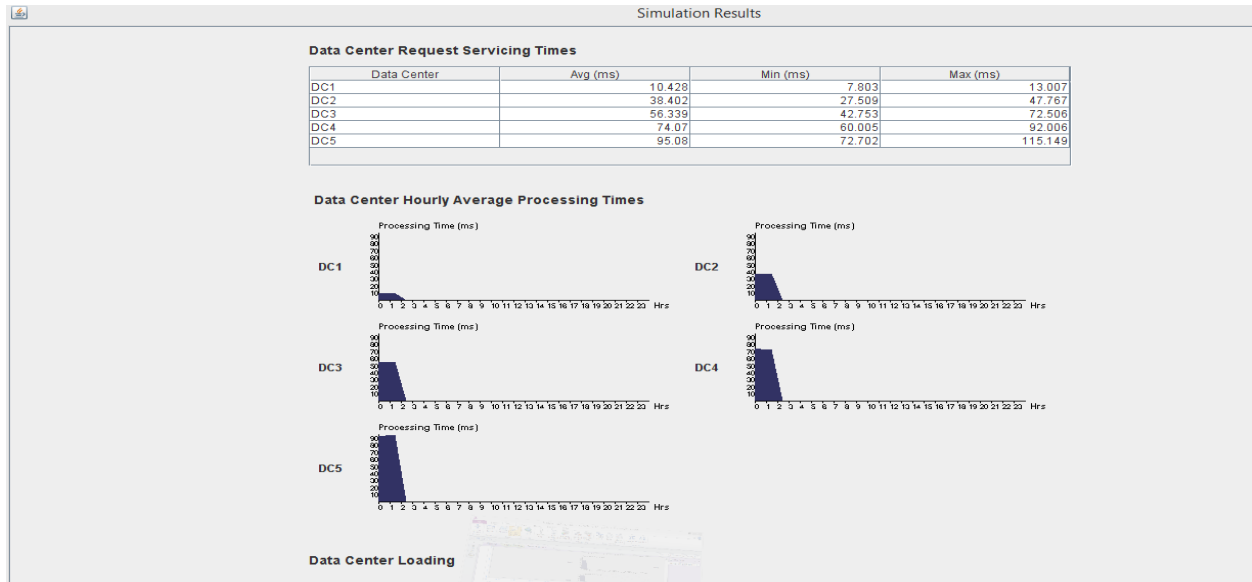


Fig 8: Data center processing time for AMLBA using different processors



Fig 9: Data Center Processing Time for MAMLBA Using Different Processors

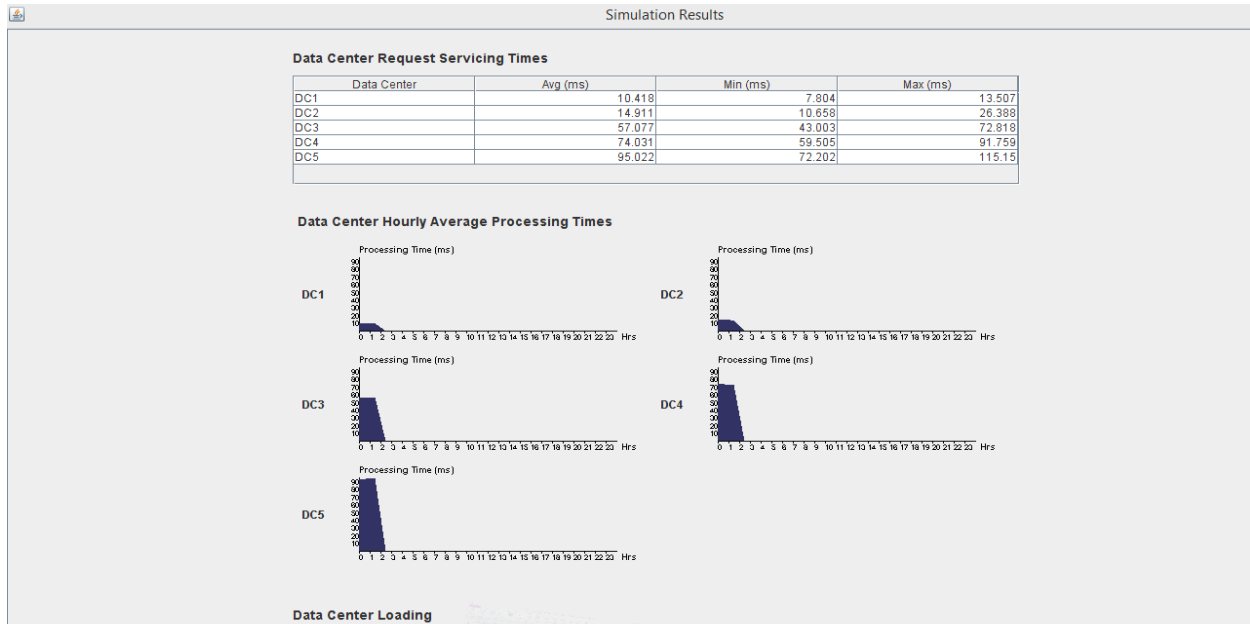


Fig 10: Data Center Processing Time for AMLBA Using Same processors

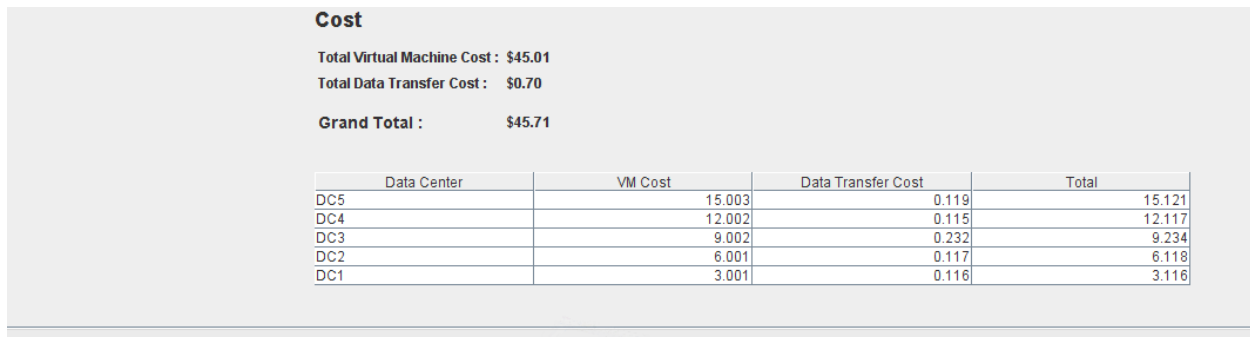


Fig 11: Cost of Load Balancing Algorithm