

# **A New Scheduling Algorithm for Workflows in Cloud Computing**

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

**Master of Engineering**  
in  
**Software Engineering**

*Submitted By*  
**Harinder Pal Singh Mahi**  
**(801231010)**

Under the supervision of:  
**Dr. Inderveer Chana**  
Associate Professor



COMPUTER SCIENCE AND ENGINEERING DEPARTMENT  
THAPAR UNIVERSITY  
PATIALA – 147004


**December 2014**

## CERTIFICATE


---

I hereby certify that the work which is being presented in the thesis entitled, "*A New Scheduling Algorithm for Workflows in Cloud Computing*", in partial fulfillment of the requirements for the award of degree of Master of Engineering in *Software Engineering* submitted in Computer Science and Engineering Department of Thapar University, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. Inderveer Chana* and refers other researcher's work which are duly listed in the reference section.

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

  
(Harinder Pal Singh Mahi)

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

  
(Dr. Inderveer Chana)  
Associate Professor,  
CSED,  
Thapar University,  
Patiala.

Countersigned by

  
(Dr. Deepak Garg)

Head

Computer Science and Engineering Department

Thapar University

Patiala

  
(Dr. S. S. Bhatia)

Dean (Academic Affairs)

Thapar University

Patiala

## ACKNOWLEDGEMENT

---

There are no words in which I can express my gratitude and respect towards my guide, Dr. Inderveer Chana. I am very thankful for her help and support in completing my thesis work.

I am also thankful to Dr. Deepak Garg, Head of Computer Science and Engineering Department, for the help and cooperation needed in completion of this thesis.

I would also like to thank all the staff members and classmates for their priceless suggestions and sharing their expertise which helped me to accomplish this work.

I am also thankful to my parents for supporting my decisions and helping me to achieve my goals.

*Harinder Pal Singh Mahi*

**Harinder Pal Singh Mahi**

**(801231010)**

## ABSTRACT

---

In the field of Information and Technology, Cloud Computing is a new trend which provides various services demanded by users. It provides the services on the basis of “pay as you use” model by providing virtualized resources with the help of internet. It has various characteristics like scalability, multi-tenancy, high performance, etc. which makes it capable to host applications related to business, social and scientific work.

The various fields of working with Cloud system are increasing day by day. But at present there are many challenges related to energy management, server consolidation, automated provisioning of services, Virtual resource migration, security, etc.

This research mainly focuses on the problem of scheduling the tasks in cloud computing. The various existing scheduling techniques focusing on several parameters improving QoS have been analyzed. They mainly focused on minimizing the time of execution and are unaware of the utilization of resources. Thus a more efficient approach for scheduling tasks is proposed to minimize the completion time and maximizing the utilization of various available resources. The proposed approach has been validated with help simulation.

# TABLE OF CONTENT

---

---

<b>Certificate.....</b>	<b>i</b>
<b>Acknowledgement.....</b>	<b>ii</b>
<b>Abstract.....</b>	<b>iii</b>
<b>Table of Contents.....</b>	<b>iv</b>
<b>List of Figures and Table.....</b>	<b>vi</b>
<b>Chapter 1: Introduction.....</b>	<b>1</b>
1.1 Cloud Computing.....	1
1.2 Cloud Service Models.....	2
1.3 Cloud Computing Deployment Model.....	4
1.4 Characteristics of Cloud Computing.....	5
1.5 Research Issues in Cloud Computing.....	6
1.5.1 Energy management.....	6
1.5.2 Server Consolidation.....	6
1.5.3 Automated Service Provisioning.....	6
1.5.4 Virtual Machine Migration.....	6
1.5.5 Security Issues.....	6
1.5.6 Workflow Management Issues.....	7
1.5.7 Traffic Management and Analysis.....	7
1.5.8 Novel Cloud Architecture.....	7
1.6 Organization of the Thesis.....	8
<b>Chapter 2: Literature Review.....</b>	<b>9</b>
2.1 Workflow Scheduling.....	9
2.2 Existing Scheduling Techniques.....	10
2.3 Simulators.....	14
<b>Chapter 3: Problem Description.....</b>	<b>15</b>
3.1 Gap Analysis.....	15
3.2 Problem Statement.....	15

3.3 Problem Description.....	16
3.4 Objective.....	17
<b>Chapter 4: Proposed Solution.....</b>	<b>18</b>
4.1 Proposed Scheduling Algorithm.....	18
4.2 Description of Algorithm.....	18
4.3 Conclusion.....	22
<b>Chapter 5: Implementation and Results.....</b>	<b>23</b>
5.1 Tools for Cloud Environment.....	23
5.1.1 CloudSim.....	23
5.1.2 NetBeans.....	25
5.2 Results.....	25
5.3 Conclusion.....	33
<b>Chapter 6: Conclusion and Future Work.....</b>	<b>34</b>
6.1 Conclusion.....	34
6.2 Future Scope.....	34
<b>References.....</b>	<b>35</b>

## LIST OF FIGURES AND TABLES

---

---

### FIGURES:

Figure 1.1: Evolution of the Cloud Computing.....	1
Figure 1.2: Cloud Service Models.....	3
Figure 1.3: Cloud Deployment Models.....	4
Figure 5.1: Layered CloudSim Architecture .....	24
Figure 5.2: Comparison of completion time and idle VMs in case 1.....	26
Figure 5.3: Comparison of completion time and idle VMs in case 2.....	28
Figure 5.4: Comparison of completion time and idle VMs in case 3.....	30
Figure 5.5: Comparison of completion time and idle VMs in case 4 .....	31
Figure 5.6: Completion time.....	32
Figure 5.7: Idle VMs time.....	33

**TABLES:**

Table 2.1: Existing Scheduling Techniques/Approaches .....	10
Table 3.1: Gaps in existing techniques.....	15
Table 3.2: Job of 12 tasks with same execution time.....	16
Table 3.3: VMs allocation while tasks have same time.....	17
Table 4.1: Job of 12 tasks .....	18
Table 4.2: Tasks scheduled.....	21
Table 4.3: Status of Idle VMs.....	21
Table 5.1: Job of 12 tasks with same time .....	25
Table 5.2: Results of case 1.....	26
Table 5.3: Job of 12 tasks having different time.....	27
Table 5.4: Results of case 2 .....	27
Table 5.5: Job of 25 tasks having different time .....	28
Table 5.6: Results of case 3.....	29
Table 5.7: Results of case 4.....	30
Table 5.8: Completion time of various schedulers.....	31
Table 5.9: Idle VMs time.....	32

# Chapter 1

## Introduction

This chapter tells about the basics of the Cloud Computing, its evaluation and background along with the characteristics and its various types. The various research issues related to Cloud Computing have been discussed in this chapter and the structure of this thesis is also given at the end of the chapter.

### 1.1 Cloud Computing

Cloud Computing [1] is a new paradigm which aims at deploying services with the help of the internet and has become a significant technology trend nowadays. It is a model that helps in providing better on-demand and convenient access to the shared computing resource pool which can be provisioned rapidly and can be left with small effort. Figure 1.1 shows the evolution of cloud computing through various phases.

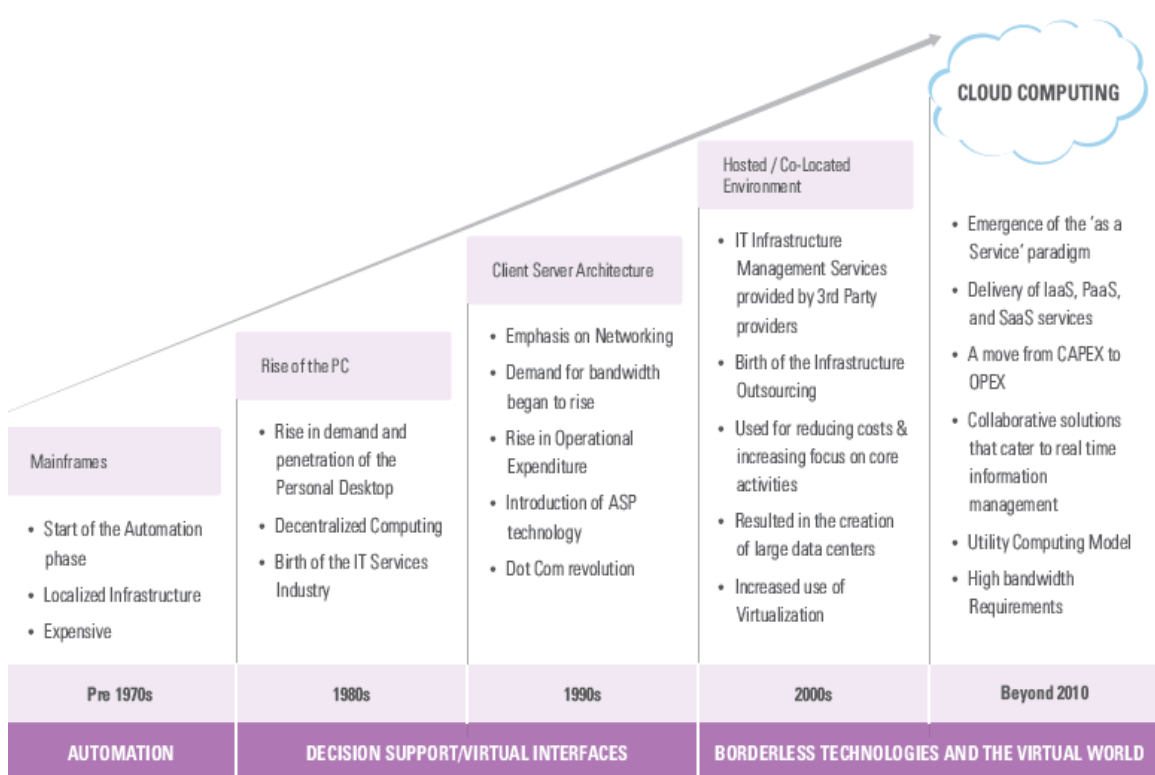


Figure 1.1: Evolution of the Cloud Computing [2]

According to National Institute of Standards and Technology (NIST) [3], Cloud Computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management efforts or service provider's interaction. The Cloud model mainly consists of two elements i.e. a front-end and a back-end. The front-end and the back-end are mainly connected via internet. The front end mainly consists of the client side computer, the network of the enterprise, and the various applications used in accessing the cloud. The back end which is the cloud provides the various resources like software application, server, and database that create a cloud of the services [4]. The various advantages of using Cloud Computing [5] are described below:

- Cloud services can be accessed easily using the internet.
- Highly economical for the users as there is less investment and running costs.
- Infrastructure and computational power is provided according to the demands from the user.
- More resource utilization which helps in increasing flexibility and elasticity.
- Reduced maintenance and business risk expenses.

## 1.2 Service Models

There are mainly three basic service models of cloud computing: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) as shown in the Figure 1.3. The different services are provided by the service providers or hosts at backend and user is can only use the services without knowing the internal working of the system.

- **Infrastructure as a Service (IaaS)** - it refers to the delivery of ondemand provisioning of infrastructural resources to the users. Users can request resources according to their need and can return them after completion of their work. The examples of IaaS are Amazon EC2, GoGrid which provides the computational and storage services. The advantages of IaaS are ability to decrease or increase the power according to demand, reduction in costs by providing hardware services, more resource utilization powers [6].



Figure1.2: Cloud Service Models [7]

- **Platform as a service (PaaS)** - it provides the various resources of the platform layer which are used for developing, testing and operating the SaaS services. The common examples of PaaS are application development engines like Microsoft Windows Azure, Google App Engine, etc. The advantages of PaaS are users only concentrate on coding and they use pay as use model for developing applications [6].
- **Software as a Service (SaaS)** - it refers to providing the services on the demand of users. Here multiple users can get services from a common source. The common SaaS providers are Microsoft's online share point, Salesforce etc. the advantage of SaaS is that users do not have to buy the software but they can use it on the cloud according to the charges on pay as you use basis [6].

### 1.3 Cloud Computing Deployment Model

There are mainly four types of deployment models in Cloud Computing as shown in the figure 1.2 namely private, public, community and hybrid cloud [8].

- **Private Cloud**

In the private cloud, the computing resources are managed by a single enterprise. It is generally deployed on the enterprise's datacenter and managed by the internal service provider. Its main advantage is that the security, compliance and quality of service are under the control of single organization.

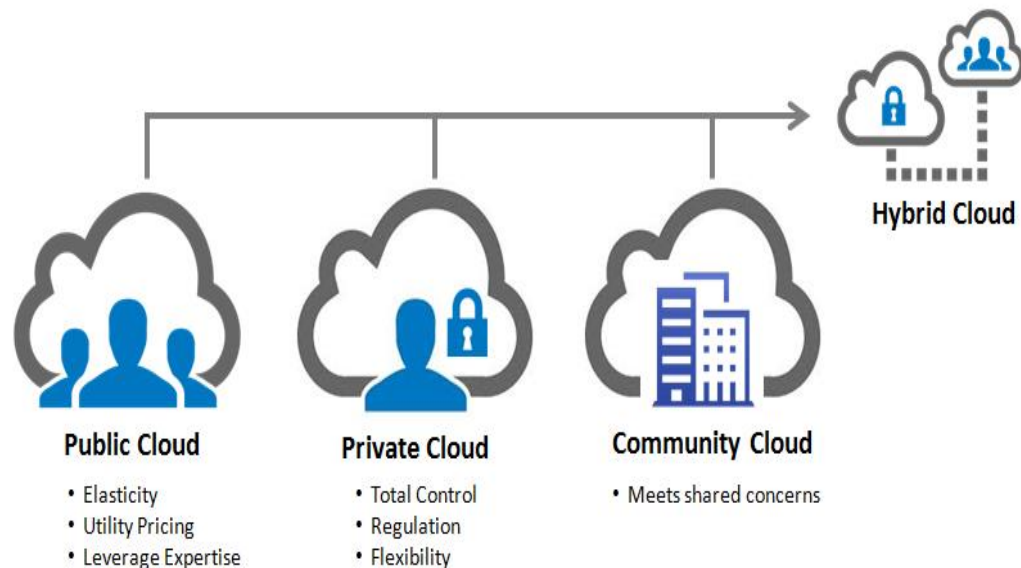


Figure 1.3 Cloud Deployment Models [9]

- **Public Cloud**

In the public cloud, the various services are provided by cloud provider to the users where cloud provider has the full control on these clouds. Its main advantage is that there is no initial capital investment on infrastructure and less data transfer risk.

- **Community Cloud**

In community cloud, the various resources, infrastructures, policies and requirements of one cloud are shared between different communities or institutions. This is different from public cloud in the way they serves multiple users and from private cloud that serves within in a particular enterprise. Its main

advantage is that it is economically scalable and the infrastructure can be managed by a third party.

- **Hybrid Cloud**

It is the combination of two or more clouds (public, private or community) and is used by various organizations. It enables the organization to run state-steady workflow in private cloud, and uses the public cloud for intensive computing resources when peak workload occurs, and returns back to its default state. The main advantages of using hybrid cloud are its more flexibility, security than other cloud models. It provides on demand service expansion and compaction whenever needed.

## 1.4 Characteristics of Cloud Computing

The various characteristics of the Cloud Computing model are [6] [10]:

- **Multi-tenancy** - services owned by different providers are located on same datacenters in the cloud environment.
- **Utility based pricing** - Cloud Computing provides various services on the pay per use pricing model. The pricing may vary from one service to another.
- **Dynamic resources provisioning** - dynamic resource provisioning helps the providers to provide resources on the current demand which lowers the operation costs as compared to traditional models.
- **Self-organizing** - as the service providers have the power to manage the resources according to their needs so the situations of high resource demand can be managed easily.
- **Shared resource pool** - the infrastructure provider can dynamically provide resources to different customers at same time by virtualization and multi-tenancy.
- **High availability** - the services in cloud environment are available through internet which helps the users to access the services at any place on devices such as phones, tablets, netbooks etc.
- **Ubiquitous network access** - the services provided by the provider available on the network can be accessed by both the thin and thick clients.

- **Increased storage** – the cloud providers are also giving services for storing and maintaining large amount of data, so that users can store their data at single place.

## **1.5 Research Issues in Cloud Computing**

The Cloud Computing is an emerging field having high impact and control on the IT industry but still it is the initial phase and has many challenges in front of it. The various issues in Cloud Computing are as follows [6] [11]:

### **1.5.1 Energy Management**

The major issue in cloud environment is to reduce the energy consumption and make the cloud more energy efficient. To make this happen the service providers must use energy efficient scheduling and server consolidation so that resource wastage gets less [6].

### **1.5.2 Server Consolidation**

It is a very effective way for increasing the resource utilization by consuming less energy. The migration of live Virtual machines is done to consolidate the Virtual Machines on different working servers onto a single server which results in energy saving [6].

### **1.5.3 Automated Service Provisioning**

In cloud computing the service provider is capable of allocating and de-allocating resources to the cloud for the meeting requirements, while dropping the operational cost. This typically involves: construction of a model for application performance to predict number of instances needed at each level for handling demands for satisfying QoS, prediction of future demands using performance modeling, and the allocation of predicted resources [6].

### **1.5.4 Virtual Machine Migration**

It enables the highly robust and responsive provisioning in the data centers. It is done using virtualization technique by balancing the across the data centers. Virtual Machine is done with the help of migration techniques. The main issue is migration of the whole VM in the current state from one server to another within a few milliseconds securely and without any data loss [6].

### **1.5.5 Security Issues**

The security issues in cloud computing are mainly related with availability of service and the security of data [11].

Availability of service – the availability of services can be disturbed by Denial of Service (DoS) attacks.

Data Security – the confidentiality and the auditability plays major role in securing data in the cloud. Confidentiality can be achieved using cryptography to transfer data securely and auditability is done to verify the originality of the data.

#### **1.5.6. Workflow Management Issue**

Workflow Management is also an important issue in Cloud Computing. The business processes in an enterprise can be represented as a workflow. So the workflow can be defined as the collection of coordinating tasks designed to do a well-defined business process. The task innovation, execution, synchronization and flow of information are managed by the workflow management. The workflow scheduling is the problem of finding a better sequence of execution for the tasks in the workflow, i.e. the execution obeying the constraints of the business logic of the workflow [12].

#### **1.5.7 Traffic Management and Analysis**

Traffic management and analysis of data is very important in data centers. Many applications rely on the analysis of data for optimizing their customer experiences. There are some challenges related to this:

Density of links – it is much higher than that of service providers, which creates problem for existing methods.

Computations of hosts – the existing methods have the ability to compute traffic few hundred hosts while a modular data centre can have thousands of servers [6].

#### **1.5.8 Novel Cloud Architecture**

The majority of commercial clouds get implemented in large data centers which are operated in centralized form. Although this design achieved economic benefits but there are some limitations like high energy usage and lot of initial investment for the construction of data centers [6].

## **1.6 Organization of the Thesis**

The thesis is organized as follows:

**Chapter 2** - In this chapter the analysis of the existing scheduling techniques is done.

**Chapter 3** - In this chapter the problem definition is described along the gap analysis of existing algorithms.

**Chapter 4** - In this chapter the solution for problem is given using proposed approach.

**Chapter 5** - In this chapter the implementation details and experiment results are shown.

**Chapter 6** - In this chapter the conclusion, contribution future work is discussed.

## Chapter 2

### Literature Review

---

This chapter discusses the study about the various existing scheduling algorithms and existing simulators for the cloud environment.

#### **2.1 Workflow Scheduling**

Workflow Scheduling is the mapping and management of execution of tasks on shared resources that may not be in direct control of the system. The various features of the scheduling are:

- The scalability, quality and performance of the system depend on the scheduling architecture. The scheduling decisions for every task are done by one central scheduler. In decentralized Approach there is no controller present for multiple schedulers but the schedulers can communicate with each other. The scheduler assigns the job to the various resources having minimum load. But in the hierarchical approach a central manager is present which controls the execution and assigns the tasks to schedulers at lower levels [13].
- The scheduling decisions taken only on the basis of tasks are referred as the local decisions whereas the scheduling decisions taken on the basis whole task are referred as global decisions. The latter gives better results since the local takes scheduling decision only one task.
- Static and dynamic are two ways for transforming abstract workflow into concrete. The concrete model gets generated before execution using available information of the environment in the static. It is of two types user-directed and simulation based. In user directed decisions of mapping and scheduling of resources is done on the basis of knowledge of the user whereas in simulation based the simulation of execution of tasks is done before the actual execution of task. There are two approaches for dynamic scheme, i.e prediction based and just in time based.
- Scheduling strategies are of three types namely performance driven, trust driven and market driven. In performance driven, the main motive is to achieve high

performance for QoS parameter. In market driven, the motive is to increase availability, decrease allocation cost. In trust driven, focus is on the security and scheduling of tasks is done on trust [13].

## 2.2 Existing Scheduling Techniques/Approaches

The various existing scheduling techniques for the cloud environment are shown in the Table 2.1.

Table 2.1 : Existing Scheduling Techniques/Approaches

Author	Year	Algorithm	Parameters	Method	Tools
Meng Xu et al. [14]	2009	Multiple QoS with Multiple Workflow	Makespan	Schedules the task with reduced cost	CloudSim
S. Pandey et al. [15]	2010	Best Resource Selection(BRS)	Transmitting and computing cost	Calculating the average computation and transmission cost	JSwarm
X. Wang et al. [16]	2011	Look Ahead Genetic Algorithm	Reliability and Makespan	Based on evaluation mechanism	GridSim
Cui Lin et al. [17]	2011	SHEFT Algorithm	Makespan	Maps workflow application on number of processor	CloudSim
El-Sayed et al. [18]	2012	Extended Max-Min Algorithm	Execution Time	Schedules the smaller tasks on weak resources and larger on faster helping in reducing the waiting time and	Java 6

				execution time	
H. Fard et al. [19]	2012	Multi-objective Workflow Scheduling	Makespan, Cost ,Energy	Minimizes the optimal parameters	ASKALON
S. Ghanbari et al. [20]	2012	Priority Job Scheduling Criteria	Makespan	Analytically represent the job priorities at objective level, scheduling level and job level	Cloud environment consists of 3 resources.
A. Delavar et al. [21]	2012	Reliable Scheduling Distributed Technique	Makespan	Calculate the request time and acknowledge time in a shared mode for increasing efficiency	Java
O. M. Elzeki et al. [ 22]	2012	Heuristics based improved Max-Min	Makespan	Modifying Max-Min algorithm using its advantages	Java
Zhangjun Wu et al. [23]	2013	Hierarchical Scheduling	Makespan, Cost and Utilization	Assigns job level and resource level scheduling known as task-to-service scheduling.	SwinDeW-C
Y. Yang et al. [24]	2013	Trust based Scheduling Algorithm	Security and reliability	Verifying probability of resource failure	CloudSim

				in transmission process	
D. kang et al. [25]	2014	Cost Based Heuristic Scheduling Approach	Cost	Assigning tasks to available resources in sequence of VMs, and execute a single VM instance in parallel	Openstack
C. Ming Wu et al. [26]	2014	DVFS Technique for Cloud Datacenters	Energy	Reduces the energy consumption using dynamic voltage and frequency scaling (DVFS) technique	-
Xiaoli Wang et al. [27]	2014	Multi-objective Bi-level Programming Model	Energy	Increases the performance of servers taking into account	MapReduce

MengXu et al. [14] mainly worked on cost and makespan. In this work the mean time for execution of all the tasks is calculated and scheduling is done according to task having minimum variation in time and optimization of cost.

Suraj Pandey et al. [15] presented an approach based on the particle swarm optimization which takes into account the parameters of cost of transmitting and computing. The work of this method is to calculate the average computing cost and average transmitting cost of every task, schedule according to minimum cost found.

X. Wang et al. [16] proposed a look ahead genetic algorithm for the optimization of reliability and makespan in applications of workflow. It is optimization of a typical genetic algo according to resource priority.

Cui Lin et al. [17] proposed SHEFT algorithm (Scalable Heterogeneous Earliest Finish Time algorithm) for scheduling workflows in Cloud computing environment. It is basically an extention HEFT algorithm, which was applied to map an application of workflow to bounded number of processors.

El-Sayed et al. [18] presented an approach to overcome the limitations of existing scheduling technique Max-Min. A modified Max- Min algorithm is applied considering the advantages and covering up the disadvantages.

S. Ghanbari et al. [20] presented a priority based job scheduling, which considers the makespan for scheduling in cloud bu using three level process of analytical hierachy. A. Delawar et al. [21] presented approach reliable scheduling distributed. Its proper evaluation is done and the results are compared with PPDD approach.

Zhangjun wu etal. [23] presents work on the basis of genetic, priority, PSO approaches for scheduling on job and the reource level by taking in account the parameters like makespan resource utilization, and costs.

Y. Yang et al. [24] presented the approach where scheduling is done by checking the probability of the failure of resources while transmitting tasks.

D. kang et al. [25] proposed a novel approach for workflow scheduling which optimized the cloud environment on the basis of period based pricing to overcome the limitations of existing traditional workflow scheduling approaches on grid environment.

## 2.3 Simulators for Cloud Computing

In a cloud environment the evaluation of the workflows performance is a very difficult task. There are number of simulation tools often used to calculate their performance. The various available simulators are GridSim, CloudSim, EmuSim, etc.

**GridSim** [28] – it is used to perform simulation of the various heterogeneous resources and the other functions of the grid computing like creation of new tasks, the mapping and the managing of tasks and available resources. It can be used for the simulation of techniques focused on time and cost.

**CloudSim** [29] – it extends the working of GridSim which can be used in Cloud Computing environment consisting of cloud, Virtual Machines, etc. It mainly deals with one workload but not suitable for the scheduling of the workflows where number of tasks needs to get scheduled at same time.

**EmuSim** [30] – it gives the user's ability to provide working behavior of the application and information about the inputs. It automatically generates simulation model after collecting the data about the performance of the application.

## Chapter 3

### Problem Description

---

In the previous chapter various scheduling techniques in Cloud Computing were discussed. The objective of this chapter is to analyze the gaps found during the literature survey of existing scheduling algorithms.

#### 3.1 Gap Analysis

Based on the literature review in the previous chapter few problems in the existing techniques are identified. These problems are basically due to ignorance of QoS parameters as shown in the Table 3.1.

Table 3.1: Gaps in existing techniques

Author	Year	Ignored parameters
Meng Xu et al. [14]	2009	QoS in multiple workflows having different structures
S. Ghanbari et al. [20]	2012	Priority is given before scheduling
A. Delawar et al. [21]	2012	Time of acknowledgement and request
Y. Yang et al. [24]	2013	Trust scheduling only considering probability of failure with reliability
D. kang et al. [25]	2014	Wastage of resources leads to extra cost

#### 3.2 Problem statement

There are number of algorithms that can be compared on the basis of time for scheduling but if we consider same execution time for all the tasks then the scheduling can be done with the FCFS scheduler. So a new approach is proposed which will be helpful for such conditions considering QoS parameter and for efficient utilization of resources.

### 3.3 Problem description

Consider the problem in which all the tasks have the same execution time (let time =T) as shown in the Table 3.2

Table 3.2: Job of 12 tasks with same execution time

Parent task	Child task
R	1,2,3
1	4
2	4
3	5,6,7
7	8,9
8	10,11,12

A job having 12 different tasks having same time which can be scheduled on 3 Virtual Machines. The FCFS scheduling is implemented on this example. In the first iteration, FCFS scheduler will schedule task 1,2 and 3 at the given three Virtual Machines parallelly, it will take T time to execute these three tasks. For second iteration, task 4,5 and 6 will be scheduled to the three virtual machines parallelly, it will take T time to execute these three tasks. But for the third iteration, there is only task7 is available for execution because the tasks 8 and 9 cannot be executed at this time since their parent task has to be executed before them. So it will again take another T time to complete task 7 only, while 2 Vm's remains idle. Similarly for forth iteration, one virtual machine remains idle and again T time is consumed. Fifth iteration behaves like first schedule and it will also take T time to complete the execution of the three tasks 10,11,12. So in this manner resources are not utilized properly and a total of 5T time is required to complete the whole execution. Table 3.3 shows the process of task execution and number of idle machines

Table 3.3: VMs allocation while tasks have same time

<b>Time</b>	<b>Executable tasks</b>	<b>VMs</b>	<b>Execute</b>	<b>Idle VMs</b>
T	1,2,3	3	1,2,3	0
T	4,5,6,7	3	4,5,6	0
T	7	1	7	2
T	8,9	2	8,9	1
T	10,11,12	3	10,11,12	0

### **3.4 Objective**

The main objective of the proposed technique is to efficiently schedule different type of jobs so that overall completion time is reduced and proper utilization of available resources is done which in turn will help in reducing energy consumption and the cost of the idle resources.

This chapter gives the solution to solve the problem stated in the previous chapter using the proposed task scheduling technique. It also describes the design of the proposed algorithm.

Firstly all the executable tasks come in the list which will be allocated to the virtual machines for execution. After that the child tasks which became free were given to the virtual machines.

#### 4.1 Proposed Scheduling Algorithm

##### Steps of the approach

**Step1.** Create parent child table

**Step2.** Sort parent child table in ascending order according to number of occurrences in second column

**Step3.** Extract distinct values from first column keeping them in same order

**Step4.** Swap the values in the original order

**Step5.** Assign tasks to available VMs

#### 4.2 Description of Algorithm

The scheduling process of the proposed algorithm is explained by taking an example for scheduling 12 different tasks as shown in the figure 4.1.

Table 4.1 Job of 12 tasks

Parent task	Child task
R	1,2,3
1	4
2	4

3	5,6,7
7	8,9
8	10,11,12

The Table 4.1 clearly shows the example of job having 12 tasks. Process of first iteration is to create parent child table and after that sort parent child table according to occurrence in the second column then after that we extract distinct values from the first column. Basically the motive is which child has lesser parents those parents should execute first then after that swap in the original list according to the order of the distinct values then assign takes one by one to the idle VM.

**Steps of the Approach.**

Create parent child table

In **first iteration** task1, task2, task3 will come in list

**Step1.** Create parent child table

Parent Task	Child Task
1	4
2	4
3	5
3	6
3	7

**Step2.** After sorting second column according to less number of occurrences on top

Parent Task	Child Task
3	5
3	6
3	7

1	4
2	4

**Step3.** Extract distinct values from first column keeping them in same order

Distinct values came out to be 3, 1, and 2

**Step4.** Swap the values in the original queue

Values became 3, 1, and 2

**Step5.** Assign 3 to VM1, 1 to VM2, 2 to VM3

In **second iteration** task4, task5, task6, task7 will come in queue

**Step1.** Create parent child table

Parent Task	Child task
7	8
7	9

**Step2.** After sorting second column according to number of occurrences

Parent Task	Child Task
7	8
7	9

**Step3.** Extract distinct values from first column keeping them in same order

Distinct values came to be 7

**Step4.** Swap the values in original queue

Values became 7,4,5,6

**Step5.** Assign 7 to VM1, 4 to VM2, and 5 to VM3

In **third iteration** task6, task8, task9 will come in list

**Step1.** Create parent child table

Parent Task	Child Task
8	10

8	11
8	12

**Step2.** After sorting second column according to less number of occurrences

Parent Task	Child Task
8	10
8	11
8	12

**Step3.** Extract distinct values from first column keeping them in same order

Distinct value came to be 8

**Step4.** Swap values in original queue

Values became 8, 6, and 9

**Step5.** Assign 8 to VM1, 6 to VM2 and 9 to VM3

In **fourth iteration** task10, task11, task12 will come in the list

**Step1.** No parent child table

As there is no child present

**Step2.** Assign 10 to VM1, 11 to VM2, and 12 to VM3

Table 4.2 shows the allocation of available tasks to the given virtual machines when the execution time is same for each and every task. So in this manner resources are utilized properly and a total of 4T time is required to complete the execution of the job.

Table 4.2: Tasks scheduled

Scheduling Process	VM 1	VM 2	VM 3	Time consumed
Iteration1	Task1	Task2	Task3	T
Iteration2	Task7	Task4	Task5	T

Iteration3	Task8	Task6	Task9	T
Iteration4	Task10	Task11	Task12	T

Table 4.3 shows the number of idle Virtual Machines which in the case of proposed algorithm are zero. This table clearly shows the efficiency of new approach in reducing the resource wastage.

Table 4.3: Status of Idle VMs

<b>Time</b>	<b>Executable tasks</b>	<b>Used VMs</b>	<b>Executed tasks</b>	<b>Idle VMs</b>
T	1,2,3	3	1,2,3	0
T	4,5,6,7	3	4,5,7	0
T	6,8,9	3	6,8,9	0
T	10,11,12	3	10,11,12	0

### 4.3 Conclusion

This chapter discussed the solution to solve the problem stated in the previous chapter using the proposed task scheduling technique. It also describes the design of the proposed algorithm. Next chapter discusses the implementation part and results.

This chapter discusses about the tools and techniques used for the setup of Cloud environment. The proposed algorithm has been implemented using the CloudSim simulators and the Netbeans IDE.

#### 5.1 Tools for Cloud Environment

The different cloud applications require different types of configurations and deployment methods. The performance of these scheduling applications on real cloud environment is very challenging task. So its alternative is to create a simulation environment which can help in evaluating the performance and scheduling of the applications is an easy way. The main advantages of testing the applications in the simulation environment are:

- Simulation helps in understanding the working of system virtually
- No cost of infrastructure and services required for repeatedly testing the applications
- Evaluation of an application can be done virtually before implementing it in real cloud environment resulting in best results

The various tools used tools used for the evaluation of the proposed approach are as follows:

##### 5.1.1 CloudSim

CloudSim is a simulation toolkit which enables the modeling and simulation of the Cloud Computing environment. It also supports the modeling of the various components of the Cloud like data centers, various virtual machines (VMs). Its implementation techniques can be extended according to the application needs very easily [29].

The main advantages of CloudSim are:

- It can support the modeling and simulating a large Cloud Computing environment, including the components like data centers, various virtual machines etc.
- It contains platform for various services, provisioning and allocation policies

- It has support for connecting the simulated system components
- It takes minimum effort and time for implementing Cloud applications
- The developers can test the application performance in heterogeneous cloud environments using minimum deployment and programming.

**CloudSim architecture**

The figure 5.1 shows the architecture of the CloudSim software framework and its various components. The simulation layer provides the functions of modeling and simulation of the virtualized cloud based data centers environment having dedicates interface for managing the various VMs, memory and storage, etc.[29].

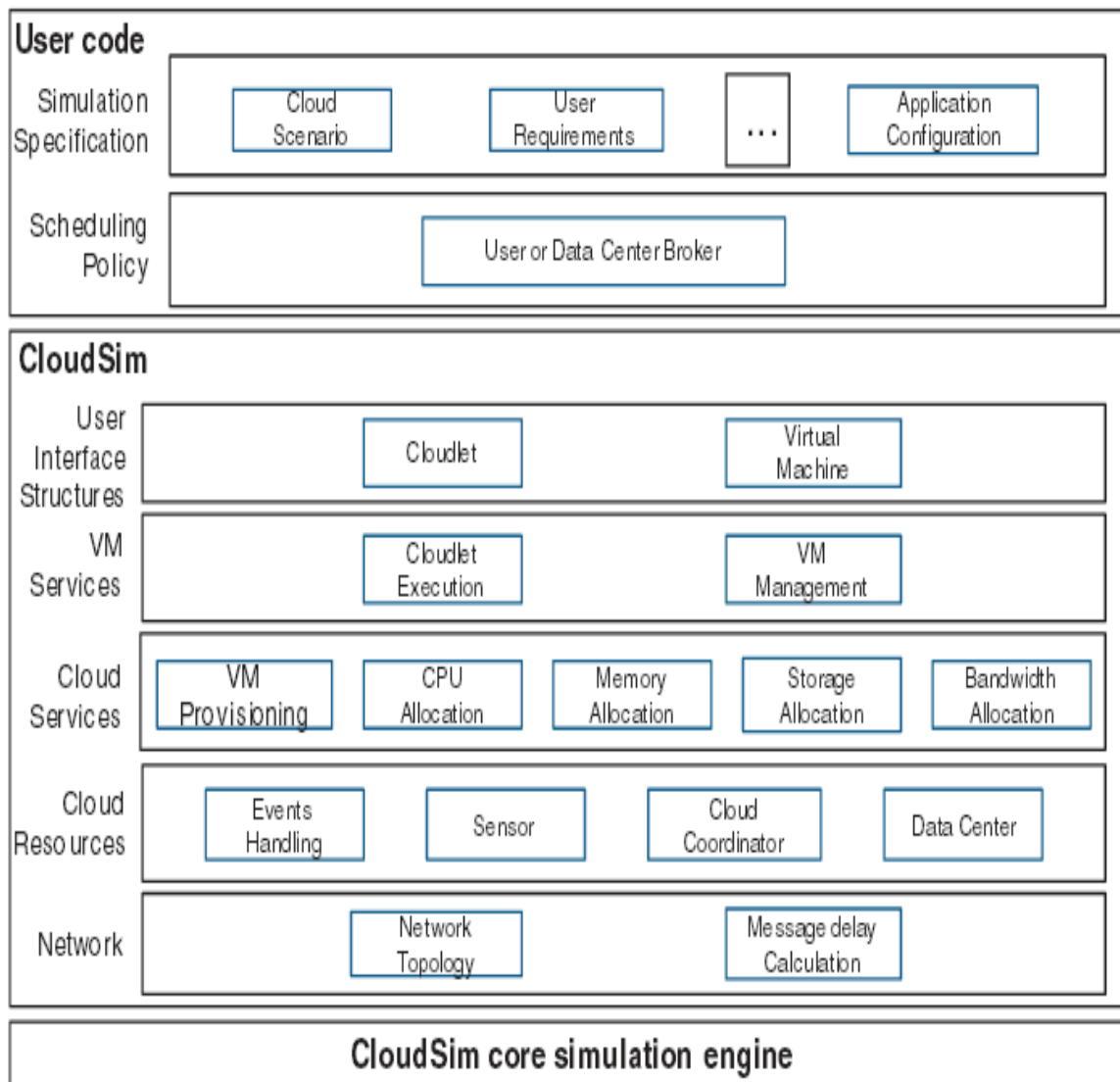


Figure 5.1: Layered CloudSim Architecture [29 ]

### 5.1.2 Netbeans

Netbeans [31] is a software development environment that can support multiple languages such as C/C++, Java, HTML, PHP, etc. In the Netbeans Integrated Development Kit (IDE) includes java development tool for the developers, it has extensible editor that helps in plug-in support for many other languages as well. The platform provides many built in library classes so there is less need for user to start from scratch.

### 5.3 Results

The implementation results have been compared with the existing scheduling algorithms like FCFS, Min-Min, Max-Min, and MCT with the proposed algorithm.

#### Case1. Job of 12 tasks of same time on 3 VMs

In case1 an example of a job of 12 tasks having same execution time using 3 virtual machines is shown in the Table 5.1.

Table 5.1: Job of 12 tasks with same time

Parent Task	Child Task
R	10,11,12
1,2	4
3	5,6,7
7	8,9
8	10,11,12

Table 5.2 shows the completion time of various schedulers, number of idle virtual machines and their idle time of case 1.

Table 5.2: Results of case 1

Approach	Completion Time	Idle VMs	Idle Time
FCFS	46.19	2	30
MIN- MIN	46.19	2	30
MAX-MIN	46.19	2	30
PROPOSED	39.19	0	0

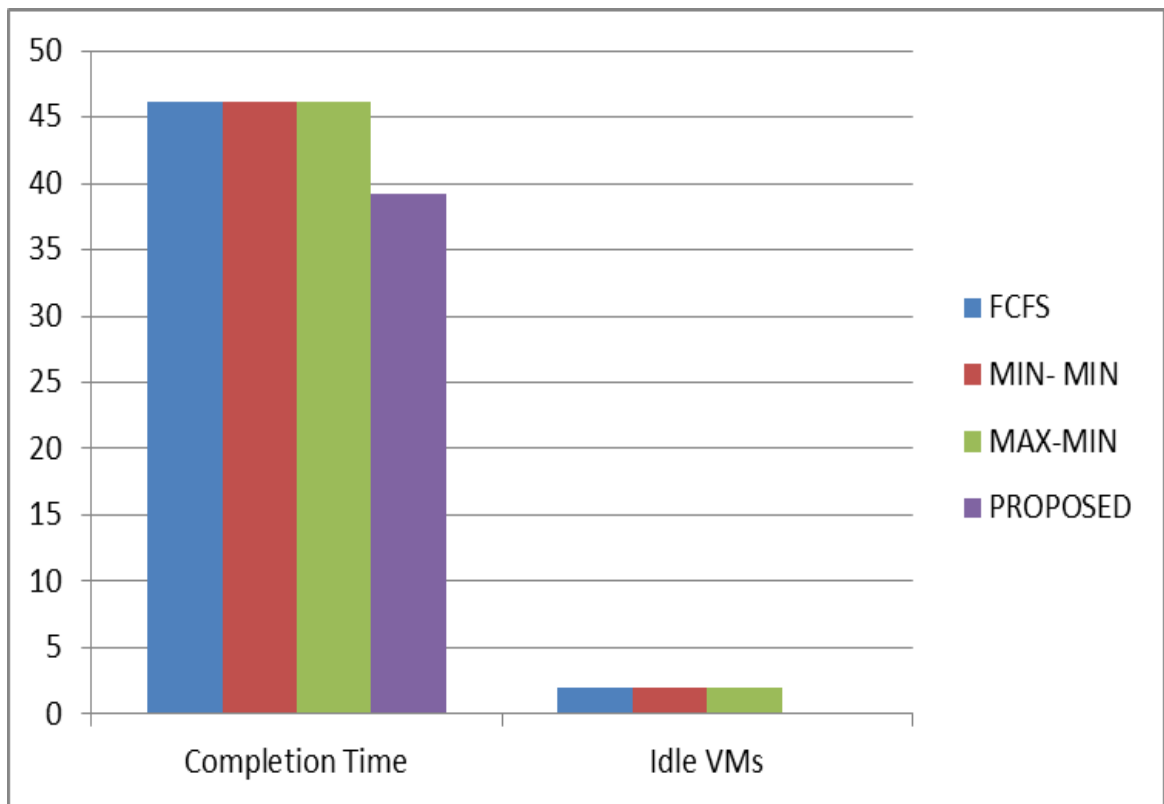


Figure 5.2: Comparison of completion time and idle VMs in case 1

Figure 5.2 shows completion time and idle VMs in case. It is clear from the figure that the completion time of the proposed approach is much less than the existing approaches and there is no idle VM.

### Case2. Job of 12 tasks having different time on 3 VMs

In case2 an example of a job of 12 tasks having different execution time using 3 virtual machines is shown in the Table 5.3.

Table 5.3: Job of 12 tasks having different time

Parent Task	Child Task
R	10,11,12
1,2	4
3	5,6,7
7	8,9
8	10,11,12

Table 5.4 shows the completion time of various schedulers, number of idle virtual machines and their idle time of case 2.

Table 5.4: Results of case 2

Approach	Completion Time	Idle VMs	Idle Time
FCFS	47.19	2	8
MIN- MIN	47.19	2	8
MAX-MIN	47.19	2	8
PROPOSED	45.19	0	5

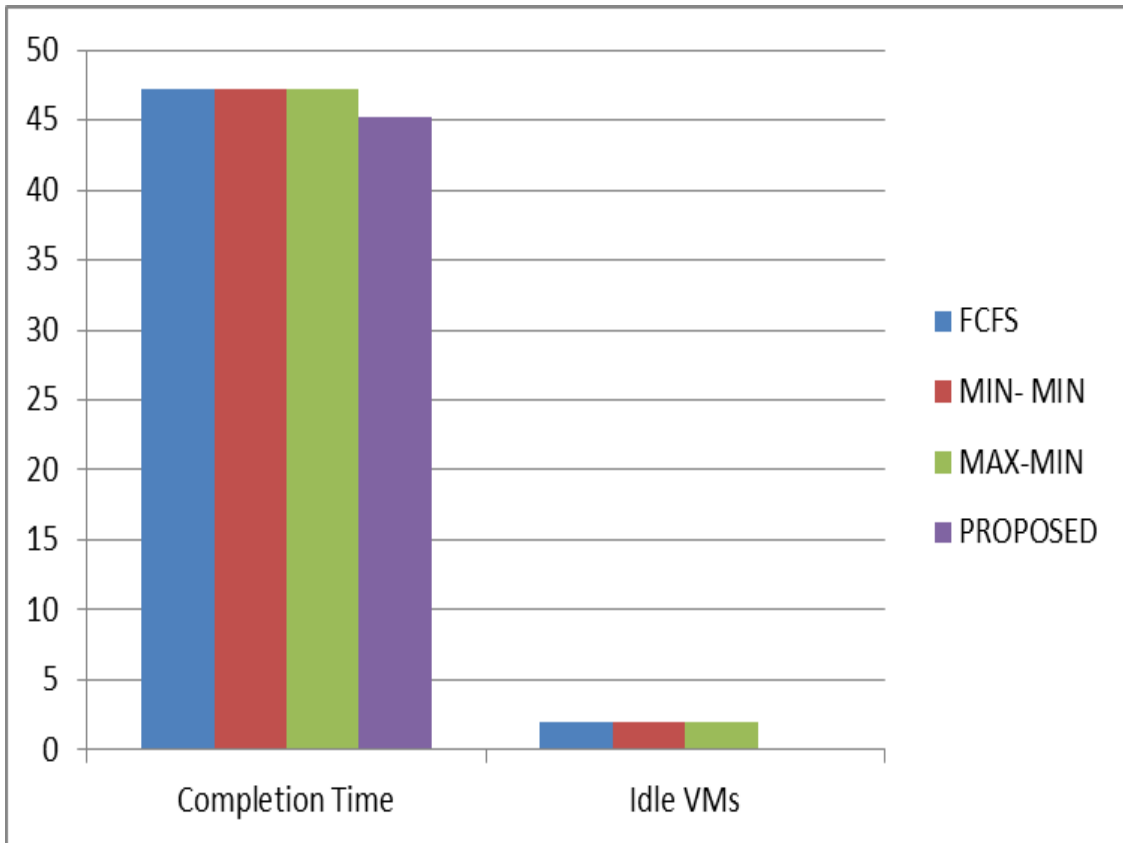


Figure 5.3: Comparison of completion time and idle VMs in case 2

Figure 5.3 shows completion time and idle VMs in case 2. It is clear from the figure that the completion time of the proposed approach is much less than the existing approaches and there is no idle VM.

### Case3. Job of 25 tasks having different time on 5 VMs

In case3 an example of a job of 25 tasks having different execution times using 3 virtual machines is shown in the Table 5.5.

Table 5.5: Job of 25 tasks having different time

Parent Task	Child Task
R	0,1,2,3,4

0	5,6,8,16
1	5,6,7,9,11,12,17
2	8,10,18
3	7,11,12,13,19
4	12,13,20
5,6,7,8,9,10,11,12,13	14
14	15
15	16,17,18,19,20
16,17,18,19,20	21
21	22
22	23
23	24

Table 5.6 shows the completion time of various schedulers, number of idle virtual machines and their idle time of case 3.

Table 5.6: Results of case 3

<b>Approach</b>	<b>Completion Time</b>	<b>Idle VMs</b>	<b>Idle Time</b>
FCFS	57.20	5	57.6
MIN- MIN	57.46	5	58.7
MAX-MIN	57.26	5	57.9
PROPOSED	57.20	5	57.6

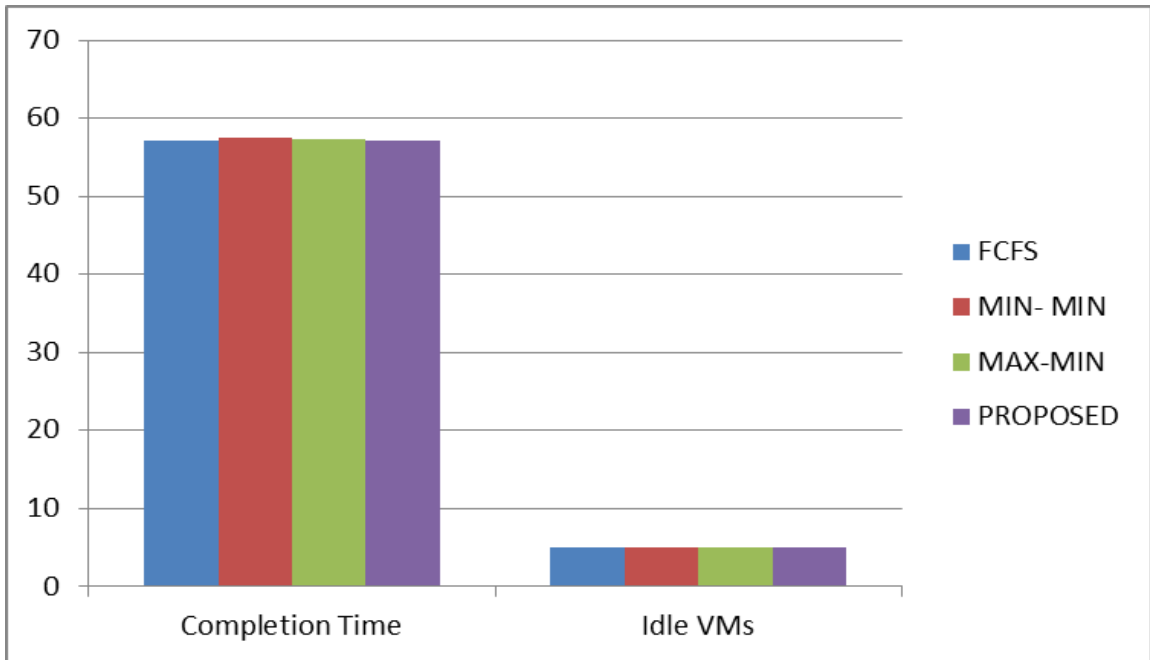


Figure 5.4: Comparison of completion time and idle VMs in case 3

Figure 5.4 shows completion time and idle VMs in case 3. It is clear from the figure that the completion time of the proposed approach is equal to FCFS approach.

#### Case4. Job of 50 tasks having different time on 5 VMs

Table 5.2 shows the completion time of various schedulers, number of idle virtual machines and their idle time of case 4.

Table 5.7: Results of case 4

Approach	Completion Time	Idle VMs	Idle Time
FCFS	129.82	5	137
MIN- MIN	133.09	5	156
MAX-MIN	129.97	5	157
PROPOSED	129.82	4	136

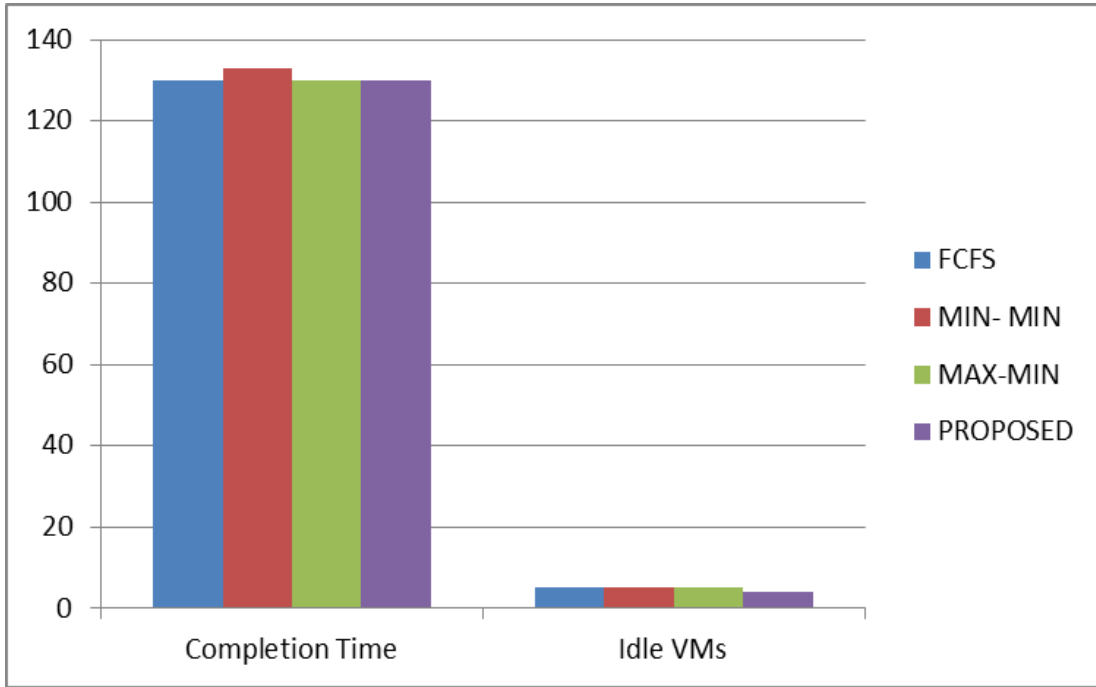


Figure 5.5: Comparison of completion time and idle VMs in case 4

The experimental results of the proposed scheduling approach are compared with existing scheduling algorithms described above using tables and graphs.

Table 5.8 shows the comparison of completion time of various algorithms. The completion time of different examples are recorded and it is clear that the completion time using proposed approach is either coming less than other approaches or nearly equal to the best completion time of existing algorithms.

Table 5.8: Completion time of various schedulers

<b>EXAMPLE</b>	<b>FCFS</b>	<b>MIN-MIN</b>	<b>MAX-MIN</b>	<b>PROPOSED</b>
Case1	46.19	46.19	46.19	39.19
Case2	47.19	47.19	47.19	45.19
Case3	57.20	57.46	57.26	57.20

Case4	129.82	133.09	129.97	129.82
-------	--------	--------	--------	--------

Figure 5.6 shows the comparison of completion time of existing techniques with the proposed technique.

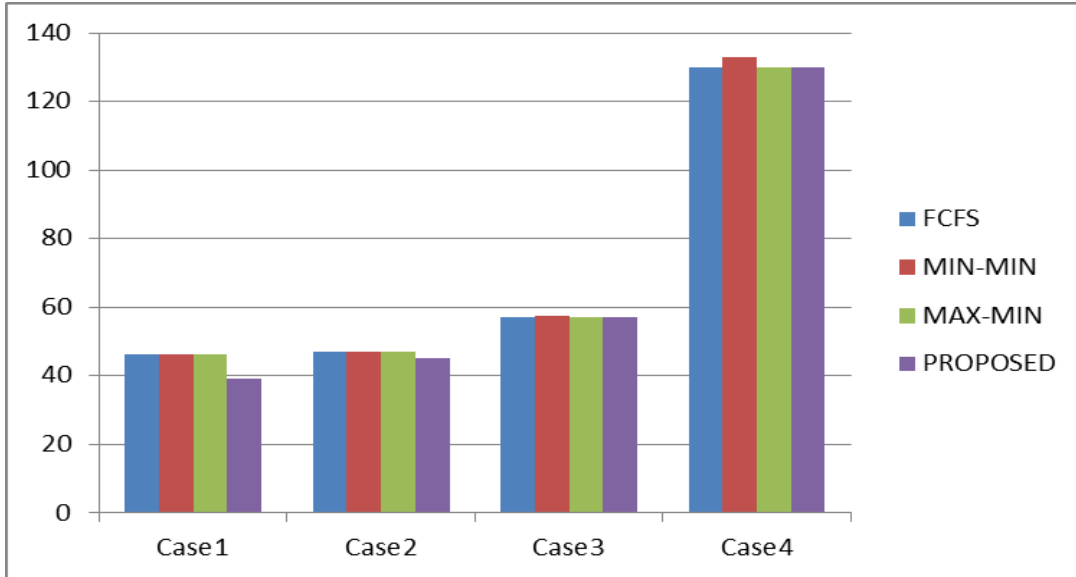


Figure 5.6: Completion time

Table 5.9 shows the comparison of the idle time of the various Virtual Machines while the execution of various tasks of the job. It is clear from the data in the table that in maximum cases the idle time of various VMs using proposed technique is either less than existing algorithms or nearly equal to the existing best scheduling technique.

Table 5.9: Idle VMs time

<b>EXAMPLE</b>	<b>FCFS</b>	<b>MIN-MIN</b>	<b>MAX-MIN</b>	<b>PROPOSED</b>
Case1	30	30	30	0
Case2	8	8	8	5
Case3	57.7	58.7	57.9	57.7
Case4	137	156	157	136

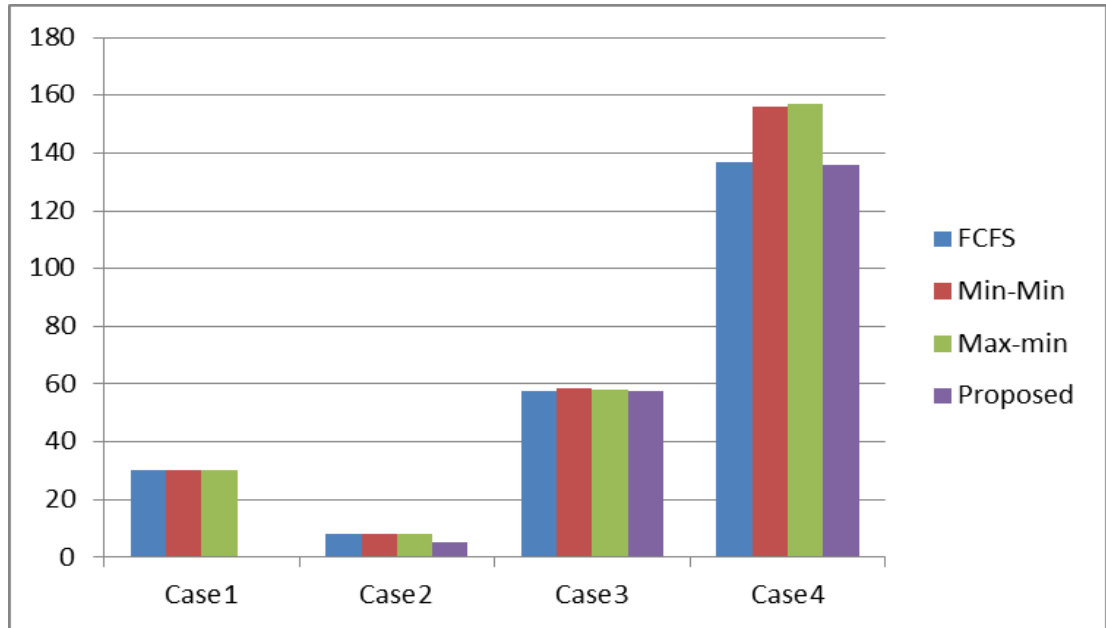


Figure 5.7: Idle VMs time

Figure 5.7 shows the time in which the Virtual Machines remain idle. It is clearly visible that the proposed approach is much efficient in reducing the idle time of the VMs.

## 5.4 Conclusion

This chapter discussed the implementation part and the results were represented in tabular and graphical form. Next chapter discusses the conclusion and future work.

## Chapter 6

### Conclusion and Future Work

---

---

This chapter summarizes the research work presented in previous chapters and also discusses its future scope.

#### **6.1 Conclusion**

The significance of this work is that it focuses on the resource utilization whereas the existing scheduling algorithms focused generally on time but not on proper resource utilization. The various gaps in the existing schedulers in cloud computing have been analyzed, on the basis of those gaps a more efficient scheduling approach is proposed. The proposed approach is proved to be quite effective to optimally use all the resources. The implementation of the proposed approach is done in simulation environment. The experimental results have shown that the proposed algorithm reduces the resource wastage and helps in efficient resource utilization.

#### **6.2 Future Scope**

At present the proposed approach is implemented in a simulated environment but in future the work can be tested on the real cloud environment. This approach is designed to reduce the wastage of resources but in future this technique can be extended by considering more QoS parameters.

## References

---

1. Buyya, Rajkumar, Chee Shin Yeo, and Srikumar Venugopal. "Market-oriented cloud computing: Vision, hype, and reality for delivering it services as computing utilities." High Performance Computing and Communications, 2008. HPCC'08. 10th IEEE International Conference on. Ieee, 2008.
2. "Evolution of the Cloud Computing" [online] Available: <http://cloudtweaks.com/2012/03/cloud-computing-leading-from-the-front-part-1/>.
3. Mell, Peter, and Tim Grance. "The NIST definition of cloud computing." (2011).
4. Jadeja, Yashpalsinh, and Kirit Modi. "Cloud computing-concepts, architecture and challenges." Computing, Electronics and Electrical Technologies (ICCEET), 2012 International Conference on. IEEE, 2012
5. Wyld, David C. "Moving to the cloud: An introduction to cloud computing in government." IBM Center for the Business of Government, 2009.
6. Zhang, Qi, Lu Cheng, and Raouf Boutaba. "Cloud computing: state-of-the-art and research challenges." Journal of internet services and applications 1.1 (2010): 7-18.
7. "Cloud Service Models" [online] Available <https://www.linkedin.com/pulse/20140907071547-305726885-iaas-pass-saas-the-cloud-101>.
8. Yang, Jianfeng, and Zhibin Chen. "Cloud computing research and security issues." Computational Intelligence and Software Engineering (CiSE), 2010 International Conference on. IEEE, 2010.
9. "Cloud Deployment Models" [online] Available: <http://transformcustomers.com/wp-content/uploads/2014/09/Cloud-Computing-Deployment-Models.png>
10. Brian, Olivier, et al. "Cloud Computing." White Paper SATW (2012).
11. Reddy, V. Krishna, B. Thirumala Rao, and L. S. S. Reddy. "Research Issues in Cloud Computing." Global Journal of Computer Science and Technology 11.11 (2011).
12. Senkul, Pinar, and Ismail H. Toroslu. "An architecture for workflow scheduling under resource allocation constraints." Information Systems 30.5 (2005): 399-422.
13. Yu, Jia, and Rajkumar Buyya. "A taxonomy of workflow management systems for grid computing." Journal of Grid Computing 3.3-4 (2005): 171-200.

14. Xu, Meng, et al. "A multiple QoS constrained scheduling strategy of multiple workflows for cloud computing." *Parallel and Distributed Processing with Applications*, 2009 IEEE International Symposium on. IEEE, 2009.
15. Pandey, Suraj, et al. "A particle swarm optimization-based heuristic for scheduling workflow applications in cloud computing environments." *Advanced Information Networking and Applications (AINA)*, 2010 24th IEEE International Conference on. IEEE, 2010.
16. Wang, Xiaofeng, et al. "Optimizing the makespan and reliability for workflow applications with reputation and a look-ahead genetic algorithm." *Future Generation Computer Systems* 27.8 (2011): 1124-1134.
17. Lin, Cui, and Shiyong Lu. "Scheduling scientific workflows elastically for cloud computing." *Cloud Computing (CLOUD)*, 2011 IEEE International Conference on. IEEE, 2011.
18. El-Sayed, T., Ali Ibraheem El-Desoky El-kenawy, and F. Mohamed. "Al-rahamawy "Extended Max-Min Scheduling Using Petri Net and Load Balancing" *International Journal of Soft Computing and Engineering (IJSCE)* ISSN: 2231-2307." Volume-2, Issue-4, September (2012).
19. Fard, Hamid Mohammadi, et al. "A multi-objective approach for workflow scheduling in heterogeneous environments." *Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (ccgrid 2012)*. IEEE Computer Society, 2012.
20. Ghanbari, Shamsollah, and Mohamed Othman. "A priority based job scheduling algorithm in cloud computing." *Procedia Engineering* 50 (2012): 778-785.
21. Delavar, Arash Ghorbannia, et al. "RSDC (RELIABLE SCHEDULING DISTRIBUTED IN CLOUD COMPUTING)." *International Journal of Computer Science, Engineering and Applications (IJCSEA)* Vol 2 (2012).
22. Elzeki, O. M., M. Z. Reshad, and M. A. Elsoud. "Improved Max-Min Algorithm in Cloud Computing." *International Journal of Computer Applications* 50.12 (2012): 22-27.
23. Wu, Zhangjun, et al. "A market-oriented hierarchical scheduling strategy in cloud workflow systems." *The Journal of Supercomputing* 63.1 (2013): 256-293.

24. Yang, Yuli, and Xinguang Peng. "Trust-Based Scheduling Strategy for Workflow Applications in Cloud Environment." P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), 2013 Eighth International Conference on. IEEE, 2013.
25. Kang, Dong-Ki, et al. "Cost adaptive workflow scheduling in cloud computing." Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication. ACM, 2014.
26. Wu, Chia-Ming, Ruay-Shiung Chang, and Hsin-Yu Chan. "A green energy-efficient scheduling algorithm using the DVFS technique for cloud datacenters." Future Generation Computer Systems 37 (2014): 141-147.
27. Wang, Xiaoli, Yuping Wang, and Yue Cui. "A new multi-objective bi-level programming model for energy and locality aware multi-job scheduling in cloud computing." Future Generation Computer Systems 36 (2014): 91-101.
28. Buyya, Rajkumar, and Manzur Murshed. "Gridsim: A toolkit for the modeling and simulation of distributed resource management and scheduling for grid computing." Concurrency and computation: practice and experience 14.13-15 (2002): 1175-1220.
29. Calheiros, Rodrigo N., et al. "CloudSim: a toolkit for modeling and simulation of Cloud Computing Environments and Evaluation of resource provisioning algorithms." Software: Practice and Experience 41.1 (2011): 23-50.
30. Calheiros, Rodrigo N., et al. "EMUSIM: an integrated emulation and simulation environment for modeling, evaluation, and validation of performance of cloud computing applications." Software: Practice and Experience 43.5 (2013): 595-612.
31. "Netbeans," [online] Available: <https://netbeans.org/features/platform/all-docs.html>