STUDY ON FUNDAMENTAL USAGE OF CLOUDSIM SIMULATOR AND ALGORITHMS OF RESOURCE ALLOCATION IN CLOUD COMPUTING

Ekta Rani [#1] , Harpreet Kaur [*2]
[#]Department of Computer Engineering Punjabi University, Patiala
*Assistant Professor, Department of Computer Engineering, Punjabi University, Patiala

Abstract: Cloud computing is an internet-based pool of heterogeneous resources. The development of Cloud is scalable and reliable to make availability of resources when required to online users. Resources are provided as a service on demand with payment done on use basis. Nowadays, for less cost and maximize resource utilization, mainly computing is done on Cloud. Cloud simulator is a collection of java classes that provides simulation of concepts of cloud computing. Cloudlet scheduling is one of the most important task in cloud computing. This paper presents fundamental study on the functions of CloudSim simulator and provides tabulated view of various scheduling algorithms used in cloud environment with their parameters and results.

Keyword: ***Cloud Computing, CloudSim, Scheduling Algorithms, Load Balancing.***

I. INTRODUCTION

Cloud computing is an emanating IT technology which permit omnipresent, on demand and user oriented access to a shared pool of computing resources through virtualization. The computing resources are storage space, networks, applications and platform services which immediately arranged by service provider with least effort. It is a shared utility scattered on internet for processing, accessing information and using hardware with features of flexible, scalability in low cost thus cloud Computing is defined as real time consignment or can say distribution of services on the internet.

Cloud is classified on the basis of Deployment Model and Service Model. From the deployment view, there are four types of Cloud Models Public, Private, Hybrid and Community Cloud. In public Cloud, a service provider makes available the resources to people on internet. Cloud provides services not open to all the groups but to a specific organization, Private Cloud induced by considering the security issue of an organization. Hybrid Cloud used where inside the company use private for reliability and for public services may allow public cloud. This model shares the cloud services to the group of organizations that uses same data or they have same area of concern to use services so a community is made known as community cloud model. IBM Blue Cloud, VM Ware, Microsoft, Rackspace, Amazon Elastic Compute Cloud are companies providing cloud Models. From the view of services, three Service Models of cloud are Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Platform as a Service. IaaS takes physical hardware components of computing like storage, network, routers and servers and offer these resources virtually to users for usage therefore also known as Hardware as a Service. In this user need not to handle the upgrade software, all these things are handling by cloud service provider. PaaS in cloud provider which provides hardware or software tools as a service to users for development of applications instead of installing it on your own computer. SaaS grant the applications that can run on cloud environment and give results. SaaS setup on internet and used by customers on local network.

Scheduling Model: Scheduling is allocation of various jobs to given resources in a given feasible time with respect to constraints given by user and also by cloud service providers. Constraints given by user are like Deadline, Budget and by service providers are making maximum profit with available resources and maximize resource utilization. By considering these constraints resources are allocated using scheduling algorithms. If user specify that job should be finish in less time but user has not specify that it is finished in this much amount of time that is deadline is not mentioned this is optimizing criteria. This is our goal of scheduling.

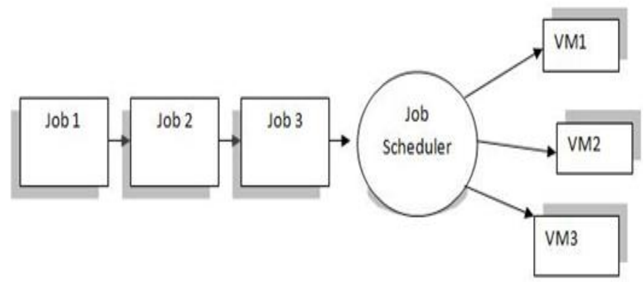

Figure 1: Scheduling Model

CloudSim: It is a stretchy framework for modeling and simulation of any application's performance in cloud environment on large scale. CloudSim is basically a library of cloud simulation scenarios. By considering policies of CloudSim, we can evaluate a method.[8] CloudSim framework includes:

Regions: Regions are geographically division to provide resources on different locations.

Datacenter: It is set of hosts or servers which provide infrastructure service. Datacenter configuration can be heterogeneous or homogeneous resources.

Hosts: It is the physical entity that is resource to tasks.

Cloudlet: It is set of requests from user for computation. This class in CloudSim designs the application services like content delivery etc.

Service Broker: it decides which VM will provide service to request.

VM allocation: These policies in CloudSim model the allocation of resources to tasks.

VM scheduler: Scheduling policies of CloudSim simulate the resources allocated to cloudlet. CloudSim virtualized servers, hosts with custom made policy for facility Virtual Machines.

For proper Utilization of VMs in Cloud, require to balance the load: **Load Balancing** is a method which makes effective use of hardware or software that is required to complete its job and using the many resources instead one may increase the throughput, reliability of resources. Categories of load balancing:

1. **Static load balancing:** In static load balancing allocation of resources is done at compile time. It is a non preemptive method. All the information related to task and resource available known before the execution of task. In this type task execution time is predicted in starting so before assigning

resource tasks are divided which reduce communication delay in such way that decrease the overall completion time.

2. Dynamic load balancing: In this load balancing decision is taken at run time thus no need to predict in advance about the running time of task, works well in distributed environment. Dynamic load balancing can be centralized or decentralized, in centralized load balancing, one single central node managing and dividing load on different servers. On the other hand, each individually builds its load vector in decentralized. Load vector contain all the information regarding load on particular component. On the basis of local load vectors, load balancing decision made.[11]

The remaining paper is organized as in Section II, the related work is illustrated after this Methodology used is described with steps in Section III. Section IV includes the Simulation done via CloudSim simulator and this is followed by Section V of analysis of results and discussion. Section VI covers the Proposed Methodology for future work and at last section VII concludes and defines Future Work.

II. RELATED WORK

[*1*] Genetic Algorithm used to distribute load in network and a novel algorithm proposed known as Dynamic well organized Load Balancing algorithm (DWOLB). Paper result shows the reduction in power consumption by proposed algorithm. [*2*]A Budget Constraint Hybrid Cloud Scheduler is a heuristic Algorithm used to maintain data privacy in workflow applications. This algorithm assigns sensitive data to private and non sensitive data to public cloud. Algorithm is good to used make satisfied budget constraint from user and minimizes makespan. [*3*] From the Ants Food hunting nature, Ant Colony Algorithm implemented which gets more response time and better quality on large and dynamic scale in cloud environment. [*4*] Algorithm gives better Performance than others. Tabu Search Algorithm utilizes all the resources effectively. Throttled Load Balancing reduces the response time when each VM works quickly. [*5*] Genetic Algorithm compared with Greedy Based, First Come First Serve Execution Time with different tasks configuration is less by Genetic Algorithm based-Search approach from First come First Serve and Greedy based. [*6*]This discusses the two major problems of task scheduling that is cost and deadline. The main goal is to find the critical path tasks to reach to deadline using Critical path based algorithm which reduces the cost for workflow applications. [*7*]This Paper modified Genetic Algorithm with mechanism of Population Reduction. This compares user requests with less response time and utilizes resources. This can be used further with different selection method for handling load on Virtual Machines.[*8*] This shows the cloud dependency on security, speed, and privacy thus point out the speed of using resources with basic load balancing algorithms. This improved and verified the resource allotment with the help of Cloud Bus in simulation Toolkit CloudSim. [*9*] In this paper, Neighbour Awareness and concept of prediction introduced. Random walk selection used on least loaded node, from neighbor according to probability node selected. With increase in probability of selection waiting time reduced. [*10*] Optimized task scheduling Algorithm compared with Max – Min, Min- Min, (RASA) Resource aware scheduling algorithm, improved Max-Min, Enhanced Max min. The distribution of the task over available resource achieves the makespan of 7.5 ms which is less than others. [*11*] This paper proposed an algorithm honey bee behavior inspired load balancing (HBB-LB), algorithm is to achieve well balanced load across virtual machines for maximizing the throughput. This algorithm set the priorities to tasks on virtual machines to minimize time and comparison is done on parameters Makespan, Response Time, Imbalance degree, number of tasks migrated honey bee outperform others. [*12*] Paper tells the problem of large number of Virtual Machine migration that makes the whole system costly and design a Genetic Algorithm used to obtain load balancing and optimized the problem of Virtual Machine migration which results the least load imbalance. [*13*]This algorithm propose a load balancing algorithm using Genetic Algorithm which simulate using CloudAnalyst Simulator and try to minimize make span of tasks in a set. The results of algorithm analysis with first come first serve (FCFS), Round robin (RR), Stochastic Hill Climbing (SHC) and the experiment is done with one to six datacenters which shows the purposed Genetic strategy best results. [*14*] Stochastic Hill climbing (SHC) compare with Round Robin and First Come First Serve Stochastic Hill Climbing gives less response time than Round Robin and first come First Serve from one to six data centers. Cloud Analyst in CloudSim used to simulate in cloud.

III. ANALYSIS ON EXISTING WORK

S No	Title	Authors and Year	Simulator	Parameters	Algorithm	Results
1	Effective resource utilization in cloud environment through a dynamic well-organized load balancing algorithm for virtual machines [*1*]	M. Vanitha, P. Marikkannu 2017	CloudSim	• Power Utilization of servers	Dynamic well organized load balancing (DWOLB)	➢ DWOLB Dynamic well organized load balancing using genetic algorithm ➢ dynamic well-organized load balancing is capable of reducing power consumption by approximately 25% in comparison with other load balancing technique
2	A Budget Constrained Scheduling Algorithm for Hybrid Cloud Computing Systems Under Data Privacy [2]	Amin Rezaeian, Hamid Abrishami, Saeid Abrishami and Mahmoud Naghibzadeh 2016	CloudSim	• Makespan • Cost	Budget Constrained Hybrid Cloud Scheduler (BCHCS) which	➢ BCHCS compared with Budget-constrained Heterogeneous Earliest Finish Time (BHEFT) and Heterogeneous Budget Constraint Scheduling (HBCS). ➢ Budget Constrained Scheduling Algorithm for Hybrid Cloud works for sensitive data and done in better makespan and less cost than other two.

3	Ant colony optimization algorithm for computing resource allocation based on cloud computing environment [*3*]	Guo Xin 2016	CloudSim	• Makespan	Genetic Algorithm (GA)	➢ Genetic Algorithm values compared with SA, Ant Colony ➢ From the simulation results, in the cloud computing provides less Time Cost per ms than others.
4	A Comparative Study on Job Scheduling Algorithm Augmenting Load Balancing in Cloud Computing [*4*]	Aswathy B Namboothir,Dr. R. Joshua Samuel Raj 2016	Cloudsim	• Response time • Throughput • Resource Utilization	Survey On Round Robin, Throttled Load Balancing, Min-min Algorithm, SHC, GA,Tabu SearchAlgorithm, Ant Colony Optimization, Honey Bee Behavior Algorithm, Join Idle queue	➢ Genetic Algorithm gives better Performance than others. ➢ Tabu Search Algorithm utilize all the resources effectively ➢ Throttled Load Balancing reduces the response time when each VM works quickly
5	A Genetic Algorithm inspired task scheduling in Cloud Computing [*5*]	Mohit Agarwal, Dr. Gur Mauj Saran Srivastava 2016	CloudSim	• Execution Time	Genetic Algorithm	➢ Genetic Algorithm compared with Greedy Based, First Come First Serve ➢ Execution Time with different task s configuration is less by Genetic Algorithm based-Search approach from First come First Serve and Greedy based.
6	Critical Path Based Scheduling Algorithm for Workflow Applications in Cloud Computing [*6*]	Jailalita, Maitreyee Dutta, Sarbjeet Singh 2016	CloudSim	• Makespan • Cost	Proposed Algorithm of critical path	➢ Proposed Algorithm compared with Min-Min Algorithm and Max-Min Algorithm ➢ Proposed algorithm achieves defined objectives and performs better in order to reduce the cost.
7	Improved Genetic Algorithm (GA) using population reduction (PR) for load balancing in cloud computing [*7*]	Ronak R Patel, Swachil J Patel, Dhaval S Patel, Tushar T Desai 2016	CloudSim	• Response Time • Consistency	Improved GA using PR	➢ Improved GA using PR gives better Response Time and Consistency with different range of resources.
8	Analysis and Improvement of Load Balancing in Cloud Computing [*8*]	Ojasvee Kaneria, R K Banyal 2016	CloudSim	• Response Time • Processing Time	Improved Round Robin	➢ Improved Round Robin performs better than existing Round Robin. ➢ It checks the availability of hosts and processing elements and accordingly assigns cloudlets.
9	Neighbour Aware Random Sampling (NARS) algorithm for load balancing in Cloud computing [*9*]	Ariharan v, Sheeja S Manakattu 2015	CloudSim	• Average Waiting Time • (WT) • Maximum WT • Standard deviation of load per node	Neighbour Aware load balancing algorithm	➢ Neighbour Aware load balancing algorithm used with two different strategies: Probability based and Least loaded node based ➢ The least loaded node strategy performs better than probability because it keep control on waiting time.
10	An Optimized Task Scheduling Algorithm in Cloud Computing [*10*]	Shubham Mittal, Avita Kata 2013	CloudSim	• Makespan	Optimized task scheduling Algorithm.	➢ Optimized task scheduling Algorithm compared with Max –Min, Min-Min, (RASA) Resource aware scheduling algorithm, improved Max-Min, Enhanced Max min. ➢ The distribution of the task over available resource achieves the makespan of 7.5 ms which is less than others.
11	Honey bee behavior (HBB)inspired load balancing(LB) of tasks in cloud computing environments [*11*]	Dhinesh Babu L.D., P. Venkata Krishnab 2013	CloudSim	• Makespan response time • Imbalance degree • No. of	Honey Bee Behavior inspired load balancing Algorithm (HBB-LB)	➢ HBB-LB with First Come First Serve, Weighted Round Robin and Dynamic Load Balancing ➢ HBB-LB results less Execution Time, Response Time and, Less degree of imbalance and no. of

				tasks migrates		tasks migrated than others.
12	Load Balancing of Virtual Machine Resources in Cloud Using Genetic Algorithm [*12*]	Chandrasekaran K, Usha Divakarla 2013	CloudSim	• Load of different physical machines under various load conditions	Genetic Algorithm	➢ Genetic Algorithm performs better than Round Robin and Greedy Algorithm under different load conditions for any applications that is CPU oriented or memory oriented.
13	A Genetic Algorithm (GA) based load balancing strategy for cloud computing [*13*]	Kousik Dasguptaa, Brototi Mandalb, Paramartha Duttac, Jyotsna Kumar Mondald, Santanu Dam 2013	CloudSim	• Response Time with input one to six data centers	Genetic Algorithm	➢ Genetic Algorithm compared with FCFS, RR local search Stochastic Hill Climbing (SHC) ➢ Genetic Algorithm gives less response time with one to six data center configurations.
14	Load Balancing in Cloud Computing using Stochastic Hill Climbing-A Soft Computing Approach [*14*]	Brototi Mondal, Kousik Dasguptaa, Paramartha Duttab 2012	Cloud Analyst in CloudSim	• Response time with one to six data centers	Stochastic Hill climbing	➢ Stochastic Hill climbing (SHC) compare with Round Robin and First Come First Serve ➢ Stochastic Hill Climbing gives less response time than Round Robin and First cone First Serve from one to six data centers.

IV. METHODOLOGY

Figure 2 shows the procedure of work.

1. Initialization of virtual machines and cloudlets: It is the first step to start the virtual machines and cloudlets. Every VM added with characteristics like VM ID, number of Processors, Memory. Similarly Cloudlets are created with ID, length, and file size.

2.Assign the cloudlets to VMs using Soft computing: Assign the Cloudlets to the Virtual machines according to their requirements example storage requirement, bandwidth etc using different software computing techniques like genetic algorithm, honey bee inspired algorithm.

2. Simulation: Start simulating the execution of Cloudlets.

3. Calculate the performance parameters: Calculate the parameters like makespan, Response time, waiting time, load and power according to objective functions.

4. Analysis: Analyses the result by comparing the parameters of different software computing.

V. SIMULATION

The Table 1 illustrates the Configuration used while creation of Cloudlets, Virtual Machine with different characteristics in simulated environment and these all implemented with Data Center created with X86 architecture and Linux Operating System used.

Starting from Basics, In Table2 first we have implemented Virtual Machines with one Data Center and created Cloudlets executed on different Virtual Machines with policy CloudletSchedulerSpaceShared. CloudletSchedulerSpaceShared implements a scheduling policy performed by a virtual machines. According to this, only one cloudlet per VM, it means others are in waiting list. Waiting is not only before execution, it even though for CPU or other data transfer while cloudlet submission.

In second experiment, as depicted by Table3, two Hosts of same configuration of two data centers are created, two cloudlets of same configurations executed on two VMs of different Configuration by MIPS. CloudletSchedulerSpaceShared used because of presence of same number of cloudlets as the number of VMs in data center. The completion time of cloudlet assigned to VM with double MIPS is less than the first one of 250 MIPS. Therefore execution time depends upon the configuration of VMs also and the load calculated on different machines is different according to their capacity of service of VMs. Third experiment, as shown by Table 4 results the four cloudlets on VMs of one host datacenter executed. 3rd and 4^{th} cloudlets are executed on same VM using Cloudlet Time Shared VM policy. Fourth experiment, as represented by Table 5 results the two cloudlets on VMs of different configurations on one host datacenter executed. In this 3^{rd} and 4^{th} cloudlets are executed on same VM using Cloudlet Space SharedVMpolicy.

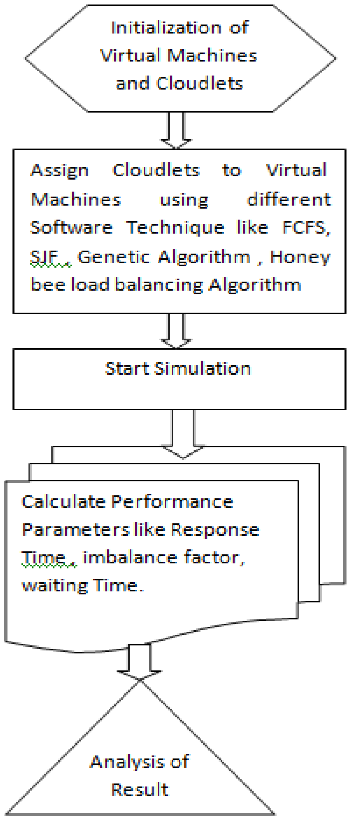

Figure 2: Procedure of work

Table 2: Creation of One Datacenter and Two Cloudlet.

Data Center	VM	Host	Cloudlet	Status	Time
1	1	1	1	Success	1000
1	2	1	2	Success	1000

Table 3: Creation of Two Datacenter and Two Cloudlet with two host

Data Center	VM	Cloudlet	Host	Status	Completion Time
1	1	1	1	Success	160
1	2	2	2	Success	80

Table 4: Creation of Two Datacenter and two Cloudlet with one Host

Data Center	Cloudlet	VM	Host	Starting Time	Finish Time	Status	Completion Time
1	1	1	1	0.1	160.1	Success	160
1	2	2	1	0.1	80.1	Success	80
1	3	3	1	0.1	106.77	Success	106.67
1	4	3	1	0.1	106.77	Success	106.67

```
Simulation completed.

========== OUTPUT ==========
Cloudlet ID    STATUS    Data center ID    VM ID    Time    Start Time    Finish Time
    0         SUCCESS        2              0       1000       0.1         1000.1
    1         SUCCESS        2              1       1000       0.1         1000.1
```

Figure 3: Screenshot of Output with one Data Center and Two Cloudlets

```
========== OUTPUT ==========
Cloudlet ID    STATUS    Data center ID    VM ID    Time    Start Time    Finish Time
    1         SUCCESS        2              1       80      0.1         80.1
    2         SUCCESS        2              2       106.67      0.1         106.77
    3         SUCCESS        2              2       106.67      0.1         106.77
    0         SUCCESS        2              0       160       0.1        160.1
*****Datacenter: Datacenter_0*****
```

Figure 4: Screenshot of Output with two Data Center and Two Cloudlets

Table 1: Configuration Used in Cloud Simulation.

S. No.	Host			Cloudlets			Virtual Machine			
	RAM (MB)	Bandwidth	Storage	Length	File Size	Output Size	MIPS For VMs	Size (MB)	RAM (MB)	CPU
1	2048	10000	1000000	40,000	300	300	250 for 2 VMs	1000	512	1

2	20480	10000	1000000	40,000	300	300	250,500 for 2 VMs	1000	2048	1
3	204800	10000	1000000	40,000	300	300	250,500,750 for 3 VMs	1000	2048	1
4	204800	10000	1000000	40,000	300	300	250,500,750	1000	2048	1

Table 5: Execution of 4 cloudlets on 3 VM

Data Center	Cloudlet	VM	Host	Status	Starting Time	Finish Time	Completion Time
1	1	1	1	Success	0.1	160.1	160
1	2	2	1	Success	0.1	80.1	80
1	3	3	1	Success	0.1	53.43	53.33
1	4	3	1	Success	53.43	106.77	53.33

VI. RESULTS AND DISCUSSION

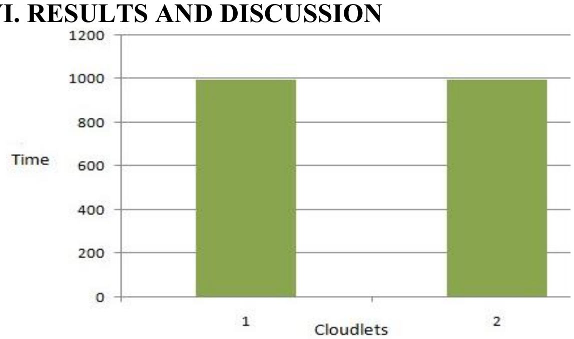

Figure 3: Execution Time of Cloudlets on 2 same VMs

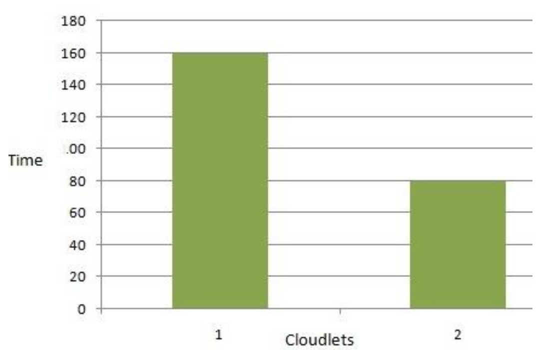

Figure 4: Execution Time of 2 Cloudlets on different Machines

Basic implementation is done in CloudSim using Eclipse. Figure 3 shows execution Time of 2 cloudlets on 2 Virtual Machines. Both Cloudlets Executed in same time the reason is same cloudlet requirement are serviced on VMs created with same configuration that same number of processor, MIPS, RAM, and Bandwidth. Second Experiment in Figure 4 results the execution Time of 2 Cloudlets on different Configured VMs that is the MIPS of Second Virtual Machine is double then the first one thus results faster completion of cloudlet assigned to second VM than the first Virtual Machine.

Figure 5 exhibits the Time of allocation of two cloudlets to two different VMs by MIPS and last 2 Cloudlets to same Virtual Machine with Time Shared Policy.

Cloudlet Space Shared Policy results the same execution time of cloudlets because both the cloudlets start their execution at same time and parallel execution because they share the time on Virtual Machine which results in late completion of Cloudlets. But Figure 6 shows the same Cloudlets are executed with same way but using Cloudlet Space Shared Scheduler Policy which outcome the reduction in Time. Using Space Shared scheduler policy only one Cloudlet per Virtual Machine Executed. So, from Figure 6 the 3rd and 4th executed on same Virtual Machine one by one that is during the execution of 3rd Cloudlet 4th one remains in Waiting state for Virtual Machine.

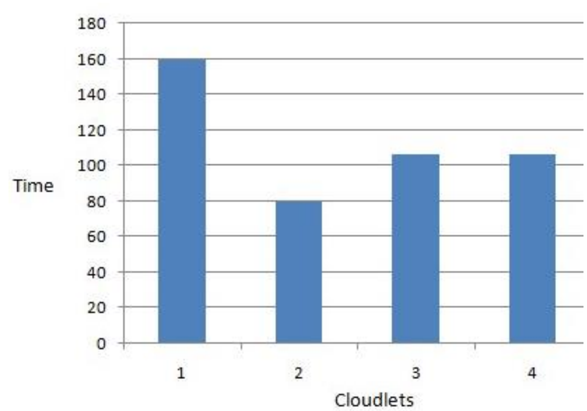

Figure 5: Execution Time of Cloudlets on VM with Time Shared Policy

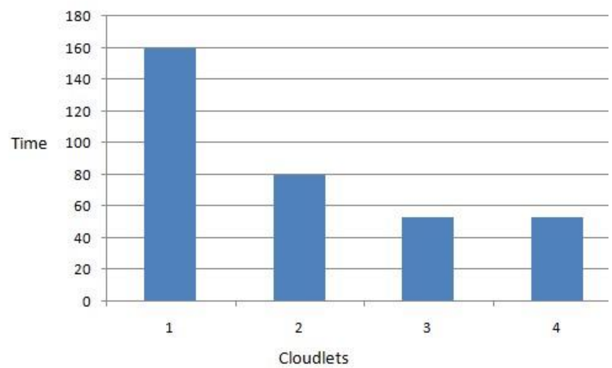

Figure 6: Execution Time of cloudlets on VM using Space Shared Policy

VII. CONCLUSION AND FUTURE WORK

This paper provides basic analysis on CloudSim Simulator, and includes simple experiments to learn to use it in Cloud Environment. In Cloud computing environment heterogeneous resources are available as services by creating Virtual Machines, these resources managed in optimized way with scheduling. Scheduling is a very essential in Cloud environment for efficient utilization. In this paper various scheduling algorithms of cloud environment based on distinguishable scheduling parameters like Scheduling factors, resource utilization, response time, time, and results have been compared and analyzed. There are some disadvantages in some scheduling algorithm, like many of algorithms minimize response time but lacks in utilization of resources, increases cost and waiting time in case of load imbalance on Virtual Machines thus need of new algorithms which considers like utilization, reliability and availability. The Above Experiments result shows the increase or decrease in completion time depending on number of tasks and policy used during allocation of resources. The limitation of space shared policy is no additional space is allocated at run time and in time shared policy average waiting time increased because resources are not released until task completed. In Future, Proposed work can be implemented with many parameters for example Completion Time with increase in load on machines, utilization of Resources in cloud Computing Environment.

Bibliography

[1] M. Vanithaa and P. Marikkannu, "Effective resource utilization in cloud environment through a dynamic well Organised load balancing algorithm for virtual machines," *Computers and Electrical Engineering*, 2017.

[2] Amin Rezaeian, Hamid Abrishami, and Saeid Abrishami, "A Budget Constrained Scheduling Algorithm for Hybrid Cloud Computing Systems Under Data Privacy," in *2016 IEEE International Conference on Cloud Engine*, 2016.

[3] Guo Xin, "Ant Colony Optimization Computing Resource Allocation Algorithm Based on Cloud Computing Environment," 2016.

[4] B Aswasthy Namboothiri and Joshua Dr. R. Samuel Raj, "A comparative study on Job Scheduling Algorithm Augmenting Load Balancing in cloud," in *2016 second International Conference on scinence Technology Engineering and Management(ICONSTEM)*, 2016.

[5] Mohit Agarwal and Dr. Gur Mauj Saran Srivastava, "A Genetic Algorithm inspired task scheduling in Cloud Computing," in *International Conference on Computing, Communication and Automation (ICCCA2016)*, 2016.

[6] Jailalita and Maitreyee Dutta, "Critical Path Based Scheduling Algorithm for workflow application in cloud computing," *2016 International Conference on Advances in Computing, Communication, & Automation (ICACCA)*, 2016.

[7] Ronak R Patel, Swachil J Patel, Dhaval S Patel, and Tushar T Patel, "Improved GA Using Population Reduction Load Bancing in Cloud Computing," *2016 Intl. Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2016.

[8] Kaneria Ojasvee and R K Banyal, "Analysis and Improvement of Load Balancing in Cloud Computing," *2016 International Conference on ICT in Business Industry & Government (ICTBIG)*, 2016.

[9] V Ariharan and Sheeja S Manakattu, "Neighbour Aware Random Sampling (NARS) algorithm for load balancing in Cloud computing," *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 2015.

[10] Subham Mittal and Avita katal, "An Optimized Task Scheduling Algorithm in Cloud Computing," in *2016 IEEE 6th International Conference on Advanced Computing*, 2016.

[11] Dhinesh Babu L.D and P.Venkata Krishna, "Honey bee behavior inspired load balancing of tasks in cloud computing environments," *Applied Soft Computing*, 2013.

[12] Chandrasekaran K. and Usha Divakarla, "Load Balancing of Virtual Machine Resources in Cloud Using," *K. R. Venugopal and L. M. Patnaik (Eds.)ICCCN2013*, 2013.

[13] Kousik Dasgupta, Brototi Mandal, Paramartha Dutta, Jyotsna kumar Mondal, and Santanu Dam, "A Genetic Algorithm (GA) based Load Balancing Strategy for Cloud Computing," *First International Conference on Computational Intelligence: Modeling Techniques and Applications (CIMTA) 2013*, 2013.

[14] Brototi Mondal, Kousik Dasgupta, and Paramartha Dutta, "Load Balancig in Cloud Computing using Stochastic Hill Climbing-A soft Computing Appproach," *ELEVIER*, 2012.