Volume 3, Issue 9, September 2013 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Modeling Local Broker Policy Based on Workload Profile in Network Cloud

Amandeep Sandhu
Student of M.tech(CSE)
Swami Vivekanand Institute Of Engineering & Technology,
Banur, Punjab, India

Maninder Kaur
Assistant Professor (CSE)
Swami Vivekanand Institute Of Engineering & Technology,
Banur, Punjab, India

Abstract— In this research paper, we have conducted work on modeling of local broker policy based on workload profile in Network cloud. For this we are using workload based applications. To handle workload based applications and distributing and scheduling broker work , two Scheduling Policies Random Non-overlap and Workload profile based policy used. We compare these two scheduling policies based on seven performance parameters Execution time (mean), Response time (mean), Waiting time (mean), Overall Sla Violation, Average Sla Violation, Execution time total standard deviation and Execution time host selection standard deviation. Workload profile based scheduling policy gave better results than Random-Non overlap scheduling policy in terms of time performance parameter.

Keywords— Cloud computing, Broker Policy , Modelling and Simulation, Data Center, Resource management, Workload Based applications.

I. INTRODUCTION

Cloud computing is the delivery of computing services over the Internet. Online file storage, social networking sites and webmail are examples of cloud services. By using Cloud computing users can use data and services from around the world in a pay-as-you-go model. Cloud Computing is like a large pool of easily usable and accessible virtualized resources, hardware and development platforms[1]. Cloud Computing offers various benefits. These benefits are On-demand self service, Provide Broad network access , Provide Resource pooling, Rapid elasticity and More Flexible, More Storage and Save Money.

In Cloud computing, the precise evaluation of scheduling algorithm for scientific applications , such as message passing parallel applications or multitier web applications modelling of data enter is required. And we found that there is no broker policy in cloud sim, which works principally on the concept of workload profile. To overcome this issue we suggest network cloud simulation. By using network cloud simulator basic workflow profile based applications have been implemented. it defines a improved method of Workload based profile. By using this method identifying and prioritize low, medium and high resource intensive cloudlets which needs to be submitted by broker to the datacenter.[2] In Cloud Computing Workload of different applications be different. Workload can vary from unrelated and independent task to related and structured workflow, which consist of sequence of connected computational or data tasks .Workload need to be managed on cloud.[3]

II. PREVIOUS WORK

Various Studies have been done on the Workflow scheduling in Cloud. Different Studies showed different results for workflow based applications. Some of them are explained below.

In a study by [2] introduced Network cloud Sim ,Which is extension of Cloud Sim. This study showed that Network Cloud Sim , was proposed. Which allows more accurate evaluation of scheduling and resource provisioning policies of a cloud infrasturucture and it also provide support for Workflow applications. The main components of Network Cloud Sim with their functionality and how different parallel applications be modeled was also defined . The evaluation results showed that Network Cloud sim is capable of simulating cloud data center and applications with communicating tasks such as MPI with high degree of reality. A Study by [4] described comparison of scheduling algorithms in cloud computing environment. This paper aim was practical comparison of four job scheduling algorithm in cloud computing. The algorithm used were Random, Round-Rubin (RR), Random Resource Selection ,Opportunistic Load Balancing and Minimum Completion Time. Three metrics for evaluating these job scheduling algorithms be throughput, makespan and the total execution cost. Based on the results, it can be also concluded that there is not a single scheduling algorithm that provides superior performance with respect to various types of quality services.

III. A REVIEW OF METHODOLOGY

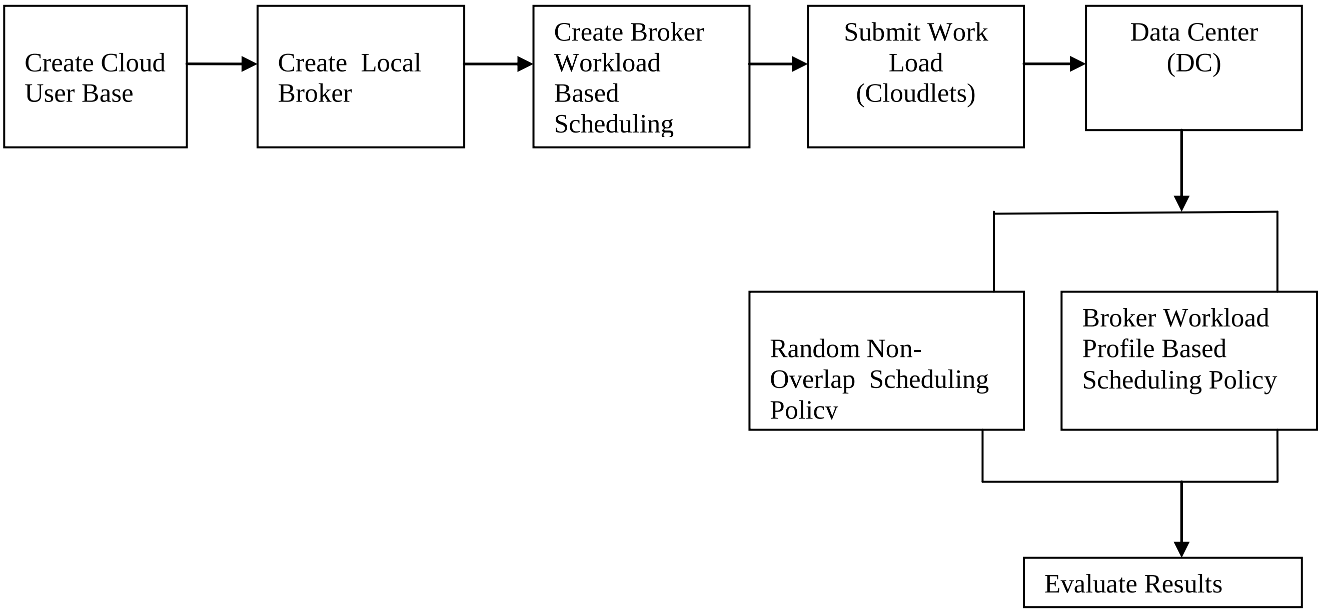
Create Cloud User Base
Create Local Broker
Create Broker Workload Based Scheduling
Submit Work Load (Cloudlets)
Data Center (DC)
Random Non-Overlap Scheduling Policv
Broker Workload Profile Based Scheduling Policy
Evaluate Results

Fig 1 Procedure for Modeling Local Broker Policy Based on Workload Profile in Network Cloud

For defining and modeling of local broker policy based on workload profile in network cloud, various steps be used. Each step briefly described below.

A. Create Cloud User Base

First step is to create a Cloud User Base.A User Base may represent thousands of users but is configured as a single unit and the traffic generated in simultaneous bursts representative of the size of the user base.[5] The User Base entities define the users of the application and their geographic distribution, and other properties such as the number of users, the frequency of usage and the pattern of usage such as peak hours.

B. Create Local Broker

After the creation of Cloud user base, we create Broker. To Create Broker it is neccsseary to understand What is Broker and What are the uses of broker.

1) What is broker :-

A cloud broker may be a third-party individual that act between the purchaser of a cloud computing service and the sellers of that service.

Local Broker:- A Local broker is one that manages the use, performance and delivery of cloud services and establish relationship between cloud service providers and cloud service consumers[6]

Global Broker:- A Global Broker system supports fast provisioning of resource infrastructures needed in service evaluation, system and computational resources, over the multiple clouds.

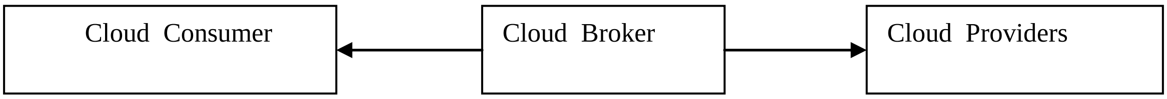

Fig 2 Role of Cloud Broker

2) Uses of Broker:-

a) The broker's main role to save the purchaser time by researching services from different vendors .

b) The broker may provide the customer an application program interface (API) and user interface (UI). By using API and UI cloud hides any complexity and allows the customer to work with their cloud services.

c) A cloud broker provide the customer with additional services.

d) A cloud broker is a software application that facilitates the distribution of work between different cloud service providers. This type of cloud broker may also be called a cloud agent.[7]

3) How to Create Local Broker:- To develop Custom defined Local broker policy , we must follow various steps that are given below. First of all initialize broker object with following properties. a)Broker Type (Local or Global) ,b)Type of Encryption c)Type Deduplication, d)Geographic Parameters , e)Number of users ,f)Definition of Peak hours ,g)Prioritization of workload.

C. *Create Broker Workload Scheduling policy:-* The third step is create broker workload scheduling policy. First we define what is workload.

1)*What is Workload (Workflow)*:- A workflow is an ordered sequence of activities or events, designed to achieve a defined business objective. In Cloud Computing different applications may result in different types of workload .Broker Policy is used for Workload profiling, Screening and Scheduling for submission to data center.

2) *Broker Workload Scheduling Policy* :- In general, scheduling is the process of mapping tasks to available resources on the basis of tasks' characteristics and requirements[8]. In this paper two scheduling policies be used. One is Random – Non overlap Scheduling Policy[2] and another one is Workload Profile Based Scheduling policy. Random non overlap scheduling policy work on FIFO(First in First out). Workload Profile Based Scheduling policy works on the principle of collecting data continuously for doing calculations related to each broker work profile. Each broker work profile contain data about broker policy. This data need to be submitted to Data center (DC).Data Center properly handle the data which is submitted by Broker policy.

D. Submit Work Load or Cloudlets:- The next step is submit work. Work be submitted to Data Center (DC). The Cloudlet class has been extended to represent a generalized task with various stages. Each stage can be computation, sending some data or receiving some data.[9]

E. Data Center (DC):- Data Center is the heart of Network Cloud. Data Center process all work, which submitted by various brokers. It consists of array hosts virtualized. Data Center has its own policy of how it will process the submitted work or cloudlets. Data center also consider broker Preferences. Data Center contains hosts , virtual machines etc.

F. Use of Two Scheduling Policies :-In this step use of two scheduling policies defined. These policies be Random Non overlap policy And Broker Workload Profile based policy. Broker Workload profile based policy was used against the Random non overlap policy. The Broker Policy is one that basically check the ratio of successful work executed without delay to the successfully executed with delay plus Urgency Number[11].
Brokers Workload Profile = Number of Successful jobs (work) without delay received / Number of successful jobs (work) with delay received + Urgency Number

Use of these two policies based on seven parameters. These parameters are Waiting Time(mean) , Execution Time(mean) , Response Time(mean) , Overall Sla Violation, Average Sla Violation, Execution time total standard deviation and Execution time host selection standard deviation

1)Execution Time(mean):- Execution time (mean) of a task is defined as the ratio of Sum of Execution Time of all tasks to total number of tasks executed.
2) Waiting Time(mean):- Waiting time(mean) of a task is defined as the ratio of sum of waiting time of a all tasks to the total number of task waiting for execution.
3) Response time(mean):- Response time(mean) of a task is defined as the ratio of sum of response time of all task to total number of tasks which response is given.
4)Overall Sla Violation:- Overall Sla violation is defined as ratio of total MIPS allocated subtracted from total MIPS requested to total MIPS requested.
5) *Average Sla Violation:*- To calculate Average Sla violation , first we calculate sla violation and then calculate its average value. Sla Violation is defined as ratio of total MIPS allocated subtracted from total MIPS requested to total MIPS requested and then calculate its average value.
6) Execution Time Total Standard Deviation:-To define the Execution Time Total Standard Deviation , we first define Execution Time Total of task and then calculate its standard deviation value. Execution time total of a given task is defined as total time spent by system execute that task , including the time spent executing run time or system services on its behalf. And the standard deviation defined as how much variation or dispersion from the average exists.
7) Execution Time Host Selection Standard Deviation:-To define the Execution Time Host Selection Standard Deviation , we first calculate Execution Time Host Selection and then calculate its standard deviation value. Execution time host selection is defined as time difference between arrival time of work and time when host is selected for executing work. And the standard deviation is defined as how much variation from average exists

TABLE I USED PERFORMANCE PARAMETERS AND ITS FORMULAS

Performance Parameters	**Formula Used**	**Charactersitics**
Execution Time(mean)	Execution Time (mean) = Sum of Execution Time of all tasks / Total number of tasks executed	Execution time ∞ 1/Performance
Waiting Time(mean)	Waiting Time (mean)= Sum of Waiting time of all tasks / Total number of task waiting for execution.	Waiting Time ∞ Queue Size of tasks

ResponseTime(mean)	Response Time (mean) = Sum of Response time of all task/ Total number of tasks for which response is given	Response time ∞ 1/ Throughput
Overall Sla Violation	Overall Sla Violation = Total MIPS Requested -Total MIPS allocated / Total MIPS Requested	Sla Violation ∞ Resouce allocation ratio
Average Sla Violation	Average Sla Violation = Average [Total MIPS Requested -Total MIPS allocated / Total MIPS Requested]	Sla Violation ∞ Resouce allocation ratio
Execution Time Total Standard deviation	Execution Time Total Standard deviation = Std dev [total time spent by system execute task +time spent executing run time]	Execution time ∞ 1/Performance
Execution time Host Selection Standard Deviation	Execution Time Host Selection Standard deviation = Std dev [Arrival time of work – time when host is selected for executing work]	Execution time ∞ 1/Performance

IV RESULTS

Since Cloud environment is a complex system which includes the matching between available computation resource and data to be processed. Efficient execution time of task is a key component of successful task scheduling and resource allocation in cloud computing environment.Before task is to be executed in data center, we are able to reduce waiting time by building efficient algorithm when work is submitted.We will doing fairly, good work in organizing overall schedule of data center. Since our algorithm work on principle of collecting data continuously for doing calculations related to each broker work profile and work submitted by each broker may not follow a linear curve ,as it may increases or decreases due to its own preferences. Urgency Number of each workload representing how urgently work is required by broker.

Broker Workload Profile = Number of Successful jobs without delay Received / Number of Successful jobs with delay Received + Urgency Number[11].

A. Use of Execution Time (mean) for Random Non Overlap and Workload Profile Based Scheduling Policies:- First we use Execution Time (mean) performance parameter for Random Non Overlap and Workload Profile Based Scheduling Policies.The value obtained be shown below in table.

TABLE II VALUE OF POLICIES BY USING EXECUTION TIME(MEAN) PARAMETER

Policies Used	**Execution Time(mean) Value in Seconds**
Random –Non Overlap Policy	20.4618
Workload Profile Based Policy	15.8701

According to the value defined in table be used, the resultant graph be shown below:

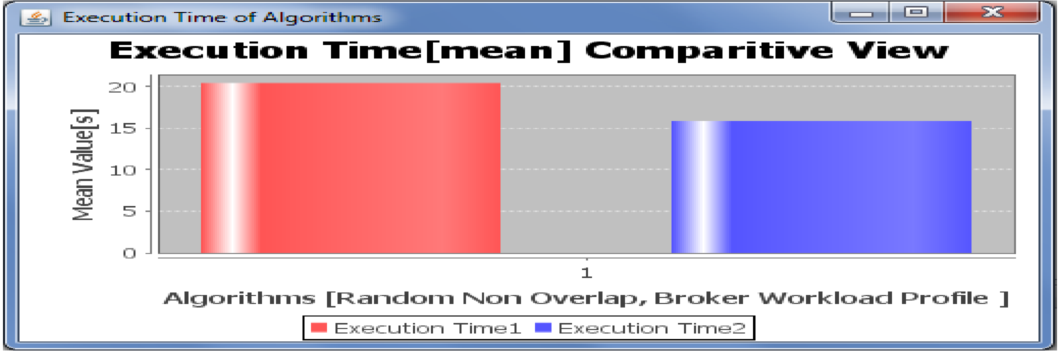

Fig 3:- Execution Time[mean] Comparative View

B. Use of Response Time (mean) for Random Non Overlap and Workload Profile Based Scheduling Policies:- Secondly we use Response Time (mean) performance parameter for Random Non Overlap and Workload Profile Based Scheduling Policies.The value obtained be shown below in table.

TABLE III VALUE OF POLICIES BY USING RESPONSE TIME(MEAN) PARAMETER

Policies Used	**Response Time(mean) Value in Seconds**
Random –Non Overlap Policy	1.2401
Workload Profile Based policy	0.7201

Value defined in table and the resultant graph be shown below.

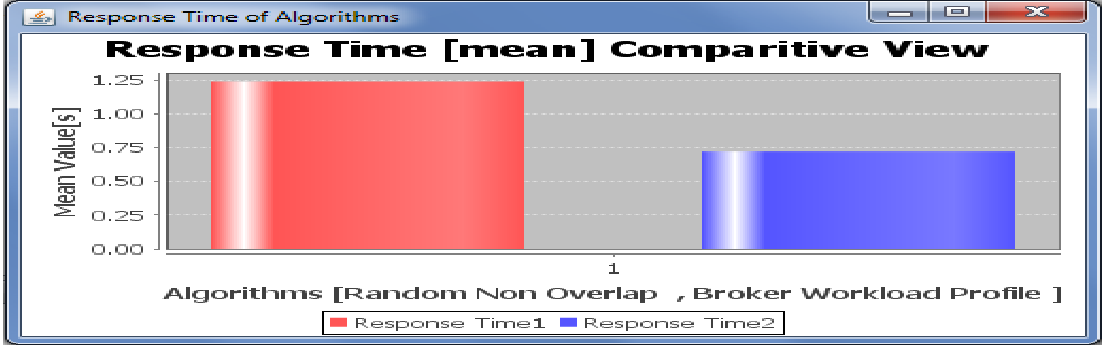

Fig 4:- Response Time[mean] Comparative View

C. Use of Waiting Time (mean) for Random Non Overlap and Workload Profile Based Scheduling Policies:- we use Waiting Time (mean) performance parameter for Random Non Overlap and Workload Profile Based Scheduling Policies.The value obtained be shown below in table

TABLE IV VALUE OF POLICIES BY USING WAITING TIME(MEAN) PARAMETER

Policies Used	**Waiting Time(mean)Value in secs**
Random –Non Overlap Policy	0.6010
Workload Profile Based Policy	0.0321

Value defined in table and the resultant graph be shown below.

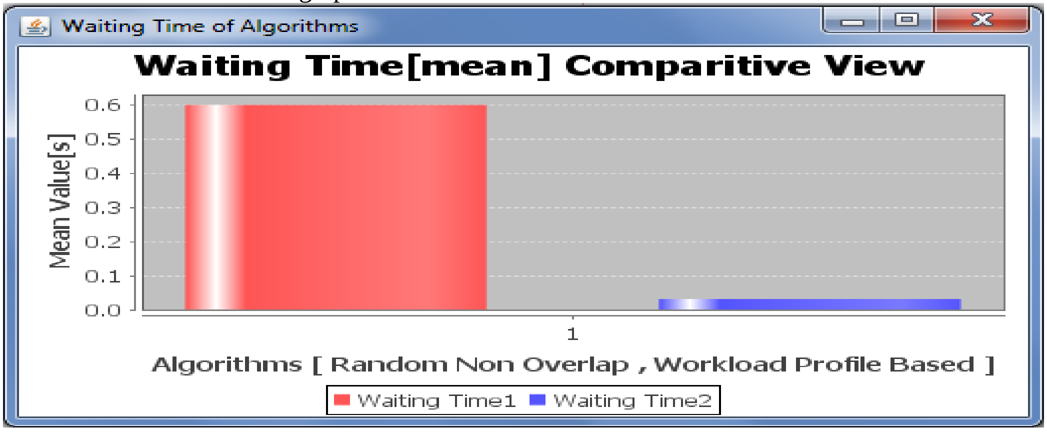

Fig 5 Waiting Time[mean] Comparative View

D. Use of Overall Sla Violation for Random Non Overlap and Workload Profile Based Scheduling Policies:- we use Overall Sla Violation performance parameter for Random Non Overlap and Workload Profile Based Scheduling Policies.The value obtained be shown below in table

TABLE V VALUE OF POLICIES BY USING OVERALL SLA VIOLATION PARAMETER

Policies Used	**Overall Sla Violation Value**
Random –Non Overlap Policy	0.014135085
Workload Profile Based Policy	0.012385219

Value defined in table and the resultant graph be shown below.

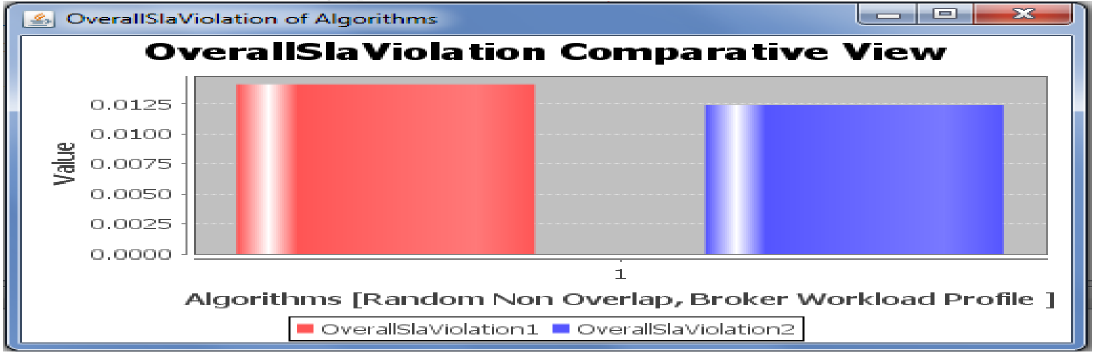

Fig 6 Overall Sla Violation Comparative View

E. Use of Average Sla Violation for Random Non Overlap and Workload Profile Based Scheduling Policies:- we use Average Sla Violation performance parameter for Random Non Overlap and Workload Profile Based Scheduling Policies.The value obtained be shown below in table

TABLE VI VALUE OF POLICIES BY USING AVERAGE SLA VIOLATION PARAMETER

Policies Used	**Average Sla Violation Value**
Random –Non Overlap Policy	0.10989566
Workload Profile Based Policy	0.108883618

Value defined in table and the resultant graph be shown below.

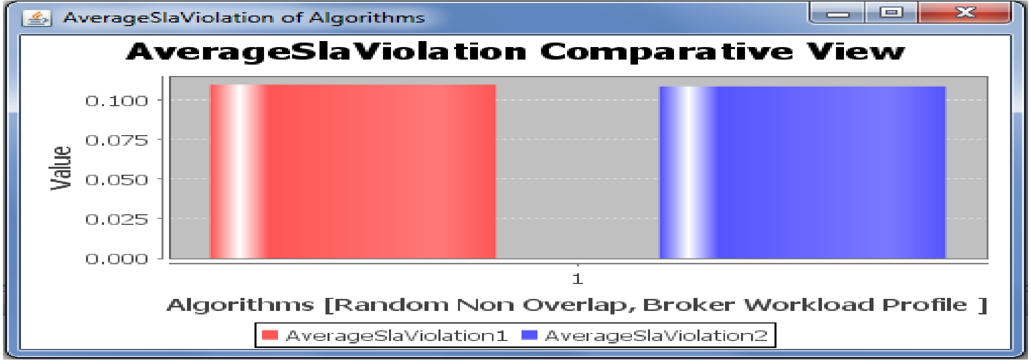

Fig 7 Average Sla Violation Comparative View

F. Use of Execution Time Total Standard Deviation for Random Non Overlap and Workload Profile Based Scheduling Policies:- We use Execution Time Total Standard Deviation performance parameter for Random Non Overlap and Workload Profile Based Scheduling Policies.The value obtained be shown below in table

TABLE VII VALUE OF POLICIES BY USING EXECUTION TIME TOTAL STANDARD DEVIATION PARAMETER

Policies Used	**Execution Time total Standard deviation**
Random –Non Overlap Policy	0.0140333333333
Workload Profile Based Policy	0.0078483333333

Value defined in table and the resultant graph be shown below.

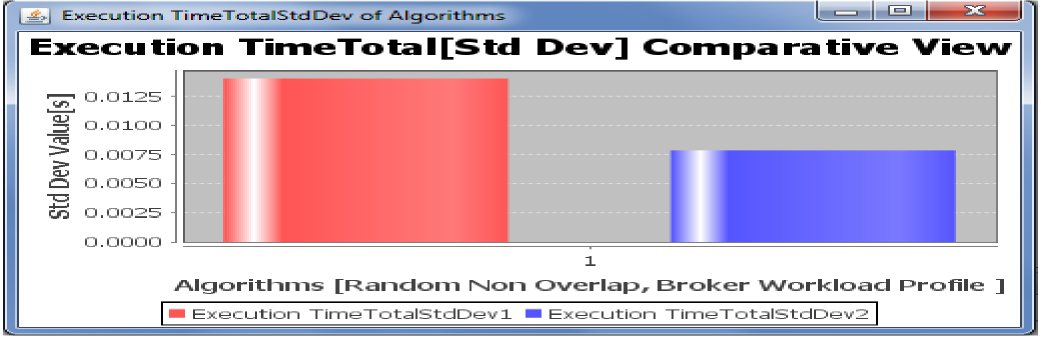

Fig 8 Execution Time Total Standard Deviation Comparative View

G. Use of Execution Time Host Selection Standard Deviation for Random Non Overlap and Workload Profile Based Scheduling Policies:- When we use Execution Time Host Selection Standard Deviation performance parameter for Random Non Overlap and Workload Profile Based Scheduling Policies.The value obtained be shown below in table

TABLE VIII COMPARISON OF TECHNIQUES BASED ON WAITING (MEAN) TIME

Policies Used	**Execution Time Host Selection Standard deviation**
Random –Non Overlap Policy	0.0009766666667
Workload Profile Based Policy	0.0009133333333

Value defined in table and the resultant graph be shown below**.**

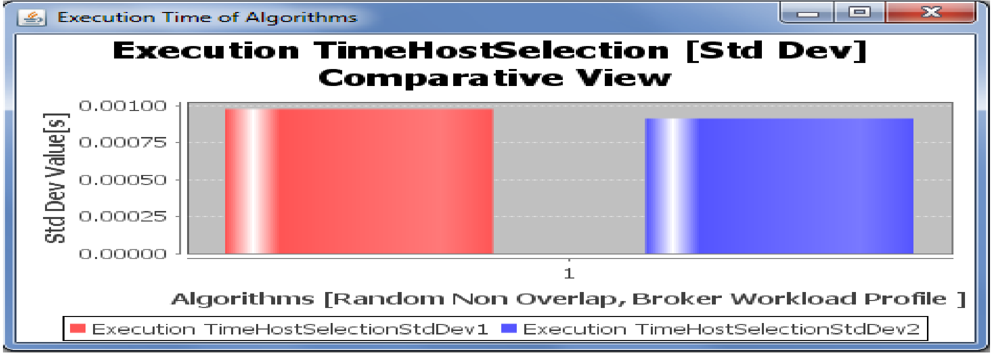

Fig 9 Execution Time Host Selection Standard Deviation Comparative View

It is apparent from the above table [2] and fig [3] that the task's execution time (is a key component of successful task scheduling and resource allocation in cloud computing environment) be less in case of workload profile policy as compared to Random Overlap policy. Now it helps to schedule the cloudlet with participation of broker policy and results which lead to reduction in waiting time of Workload profile based policy as compared to Random non overlap policy and evident from the mean values shown in the table[4] & Fig[5]. Thus this optimization helps the broker get realistic performance from the data centre based on how much and what quality of work it is submitting to the data centre, it is also evident from table[3],fig[4] the response mean value that the data centre is able to respond more efficiently and execution for data centre also reduced as more organized ,Prioritized work is received. In case of Sla violations the value of Sla violations inversely to the Resource Allocation ratio. For better performance the value of Sla violation is less. The table [5,6] and fig [6,7] showed that the value of Overall Sla violation & Average Sla violation is less for Workload profile Based policy so it gives you better performance and results as compared to Random non overlap policy. The table [7,8] and fig [8,9] showed that the value of execution time total std. dev.& execution time host selection std.dev. is less for Workload profile Based policy so it gives you better results as compared to Random non overlap policy. At the end, We conclude that by using above mentioned performance parameters , the value of Workload profile policy is less than that of random non overlap policy. The less value of workload profile based policy indicates that it is better for distributing and scheduling broker work than that of random non overlap policy.

V. CONCLUSION

To evaluate performance, we are using two scheduling policies. These are Random Non Overlap Scheduling policy and Broker Workload Profile Based Scheduling Policy. We are also use seven parameters to evaluate scheduling policy. The used seven parameters are Execution Time(Mean) ,Response Time(Mean) ,Waiting Time (Mean) ,Execution Time Total Standard Deviation, Execution Time Host Selection Standard Deviation , Overall Sla Violation and Average Sla Violation. By using these parameters we evaluate performance of two scheduling policies. When we use Random non overlap scheduling policy with respect to all parameters, its value is high. When we use Workload Profile based Scheduling policy with respect to all parameters, its value is less. The less value of workload profile based policy for all parameters indicates that Workload Profile based Scheduling policy is better than Random non overlap scheduling policy.

VI. *FUTURE SCOPE*

In our paper seven performance parameters were used. Seven parameters Execution Time(Mean) ,Response Time(Mean) ,Waiting Time (Mean) ,Execution Time Total Standard Deviation, Execution Time Host Selection Standard Deviation , Overall Sla Violation and Average Sla Violation were applied on Random Non Overlap and Workload Profile Based policies .In Future, we can use other parameters and other scheduling policies to make workload applications better.

REFERENCES

[1] CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms by Rodrigo N. Calheiros1, Rajiv Ranjan2, Anton Beloglazov1, C´esar A. F. De Rose3 and Rajkumar Buyya1.

[2] NetworkCloudSim: Modelling Parallel Applications in Cloud Simulations by Saurabh Kumar Garg and Rajkumar Bu yya.

[3] Workflow Engine for Clouds by Suraj Pandey, Dileban Karunamoorthy and Rajkumar Buyya.

[4] Comparitive Study of Scheduling Algorithms in Cloud Computing Environment by Isam Azawi Mohialdeen

[5] Cloud Analyst : A Cloud - Sim- based Tool for Modeling and Analysis of Large Scale Cloud Computing Environments by Bhathiya Wickremasinghe.

[6] "NIST Cloud Computing Reference Architecture", by Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger, Dawn Leaf NIST Special Publication 500-292, Sept. 2011

[7] A Survey of Cloud Workflow by Huang Hua 1,a, Zhang Yi-Lai 2,b , Zhang Min 3,c1,2,3 Jingdezhen Ceramic Institute ,Jingdezhen, Jiangxi

[8] Scheduling Optimization in Cloud Computing by Dr Ajay jangra , Assistant Professor Department of Computer Engg. UIET KUK , India.Tushar Saini Research scholar M.Tech Software Engg ,U.I.E.T KUK, India

[9] CloudSim: A Novel Framework for Modeling and Simulation of Cloud Computing Infrastructures and Services by Rodrigo N. Calheiros1,2, Rajiv Ranjan1, César A. F. De Rose2, and Rajkumar Buyya.

[10] Cost Effective Selection of Data Center in Cloud Environment by Manoranjan Dash, Amitav Mahapatra & Narayan Ranjan Chakraborty.

[11] Building an Infrastructure for Urgent Computings by Pete Beckman, Ivan Beschastnikh , Suman Nadellac and Nick.