



# **Comparative Study of Scheduling and Service Broker Algorithms in Cloud Computing**

Santhosh B<sup>1</sup>, Raghavendra Naik<sup>2</sup>, Balkrishna Yende<sup>3</sup>, Dr D.H Manjaiah<sup>4</sup>

Assistant Professor, Department of MCA, AIMIT, St Aloysius College, Mangalore, India<sup>1</sup>

Msc 3rd semester, Department of Software Technology, AIMIT, St Aloysius College, Mangalore, India<sup>2,3</sup>

Professor, Department of Computer Science, Mangalore University, Mangalore, India<sup>4</sup>

**ABSTRACT:** Cloud computing is a virtual pool of resources which are provided to users via Internet. It gives users virtually unlimited pay-per-use computing resources without the burden of managing the underlying infrastructure. Cloud computing service providers one of the goals is to use the resources efficiently focusing scheduling to a cloud environment enables the use of various cloud services to help framework implementation. Thus the comprehensive way of different type of scheduling and service broker algorithms in cloud computing environment is surveyed. This study gives an elaborate idea about scheduling and service broker algorithms in cloud computing. Here we have compared Service Proximity Based Routing, Performance Optimized Routing, Enhanced Proximity-Based Routing Algorithm and Resource Location and Cost based Performance Optimized Routing algorithms

**KEY WORDS:** cloud computing, scheduling, service broker algorithms

## **I. INTRODUCTION**

Cloud computing now is known as a provider of dynamic, scalable on demand, virtualized resources over the internet. It also makes it possible to access applications and associated data from anywhere. Companies are able to rent resources from cloud for storage and other computational purposes so that their infrastructure cost can be reduced significantly. They can also make use of software, platforms and infrastructures as a services , based on pay-as-you-go model.

### **1.1 Introduction to Scheduling**

Scheduling is basically allocation of given jobs to given resources in a given time period. That is how different types of jobs are allocated. Such as process has to respect constraints given by the jobs and the cloud. The kind of constraints given by users/jobs are deadline ie jobs has to be finish in given time and budgetie limited amount spend on the job. The kind of constraints given by service provider based on to maximize their resource utilization and their benefits

### **1.2 Optimization Criteria**

Optimization criteria is used when making scheduling decision and represents the goals of the scheduling process. The main objective is to increase resource usage, number of successfully completed jobs or minimize the response time. The criteria is expressed by the value of objective function which follows us to measure the quality of computed solution and compare it with different solution. Here objective function is defined with the help of constraint and criteria . When designing a scheduling algorithm our main aim is to minimize the objective function or maximize the objective function according to the criteria specified by the user.

### **1.3 Quality of Service (QOS)**

Every user wants a best quality of service for its application. QOS is the ability to provide different priority to different jobs and users or to guarantee a certain level of performance to a job. If the QOS mechanism is supported it allows the users to specify desired performance of their jobs such as Completion before given deadline, Dedicated access to the



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

resources during some time period (advanced reservation), Requested amount of resources (CPU's, RAM, HDD, Network Bandwidth) etc. Such request is formally established between the user and the resource manager (Scheduler) through negotiation which produce a formal description of the guaranteed service called Service Level Agreement(SLA).In System with limited resources the QOS support result in additional cost which is related to complexity of QOS request and the efficiency of the scheduler when dealing with them. In quality of service mechanism it allows the user to specify derived performance for their jobs. Job completion before their deadlines or dedicated access to resources this is known as advance resource reservation System.

## 1.4 Scheduling Problems

Problem of Mapping task on these resource are belong to a class of NP-hard Problems. For such problems, no known algorithms are able to generate the optimal solution within polynomial time.Solution Based on exhaustive search are impractical overhead of generating Scheduler is very high. In cloud environments scheduling decision must be made in the shortest time possible because there are many users competing for resource and time.

## 1.5 Task Scheduling

The Input of task scheduling algorithms is normally an abstract model which defines tasks without specifying the physical location of resources on which the task are executed. Mapping of job is done by the scheduler. and Reschedule When a task cant be completed due to processor failure or a disk failure or other problems. The uncompleted task could be rescheduled in the next computation.Scheduling Optimizer after acquiring information about resources through resource discovery method set of appropriate candidates in highlighted.Resource selection mechanism elects the candidates that fulfill all the requirement and optimizes the usage of infrastructure.Resource selection may be done using an optimization algorithms. Many optimization strategies may be used from simple and well know technique such as simple meta heuristics algorithm are generic algorithm. Ant colony, and practical swarm optimization (PSO) for clouds.

## 1.6 Different type of scheduling

### 1.6.1 Static and Dynamic Scheduling

In Static Scheduling jobs are pre scheduled . Information regarding as well as all the tasks in an application is assumed to be available by the time of application is scheduled and also it is assumed that no job failure and resources are assumed available all the time.

Dynamic scheduling jobs are dynamically available for scheduling over time by the scheduler with no issues, to able of determining run time in advance. Here the job execution, refer to the situation when job execution could fail due to some resource failure or job execution could be stopped due to the arrival in the system of high priority jobs. when the case of preemptive mode is considered and workload on resources can significantly vary over time.

### 1.6.2 Centralized, Hierarchical and Distributed Scheduling

Centralized and decentralized scheduling differs in the control of the resources and knowledge of the overall system.In the case of centralized scheduling, there is more control on resources .The scheduler has knowledge of the system by monitoring of the resources state. Main Advantages of centralized scheduling is ease of implementation. efficiency and more control and monitoring on resources and disadvantages are lacks scalability, fault tolerance and efficient performance. Hierarchical Scheduling allows one to coordinate different schedulers at a certain level.Schedulers at the lowest level in the hierarchy have knowledge of the resources. Main disadvantages are lack of scalability and fault tolerance, but it scales better and is more fault tolerant then centralized schedulers. Decentralized or Distributed Scheduling has no central entity controlling the resources. Scheduling decision are shared by multiple distributed schedulers and has less efficiency than centralized schedulers.

### 1.6.3 Preemptive and Non-Preemptive Scheduling

Preemptive Scheduling is the Scheduling criterion allows each job to be interrupted during execution and a job can be migrated to another resources leaving its originally allocated resources unused to be available for other jobs. It is more helpful if there are constraints as priority to be considered.



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

Non Preemptive Scheduling In which resources aren't being allowed to be re-allocated until running and scheduled job finished its execution.

## 1.6.4 Immediate vs Batch Mode Scheduling

In Immediate / Online Mode scheduler schedules any recently arriving job as soon as it arrives with no waiting for next time interval on available resources at that moment. In Batch / Offline Mode the Scheduler holds arriving jobs as group of problem to be solved over successive time intervals. So that it is better to map a job for suitable resources depending on its characteristics.

## 1.6.5 Independent and Workflow Scheduling

Independent Scheduling executes tasks independently. In Workflow Scheduling tasks are dependent on each other. Dependency means there are precedence orders existing in tasks, that is, a task cannot start until all its parent are done. Workflow are represented by Directed Acyclic Graph(DAG) notation. Each task can start its execution only when all preceding task in DAG are already finished.

## II. INTRODUCTION TO CLOUD APPLICATION SERVICE BROKER

The traffic routing between User Bases and Datacenters is controlled by a Service Broker that decides which Datacenter should service the requests from each user base. We have selected the following algorithms for our study

i). Service Proximity based routing.

In this case the proximity is the quickest path to the datacenter from a userbase based on network latency. The service broker will route user traffic to the closest datacenter in terms of transmission latency.

ii). Performance Optimized routing

Here the Service Broker actively monitors the performance of all datacenters and directs traffic to the datacenter it estimates to give the best response time to the end user at the time it is queried.

iii). Dynamically reconfiguring router

This is an extension to Proximity based routing, where the routing logic is very similar, but the service broker is entrusted with the additional responsibility of scaling the application deployment based on the load it is facing. This is done by increasing or decreasing the number of VMs allocated in the datacenter, according to the current processing times as compared against best processing time ever achieved.

iv) Enhanced Proximity-Based Routing Algorithm[14]

This is an extension to Proximity based routing. The location of the resource and cost per processing ie cost per Vm\$/Hr is considered. In the algorithm Location of the resource based on SLA is considered and when there is more than one datacenters are available it selects the data center with the lowest cost based on cost per Vm\$/Hr and also it manages the load between the datacenters. When there are more than one datacenters with the same cost per Vm\$/Hr is found it selects one of the datacenter randomly.

v) Resource Location and Cost based Performance Optimized Routing[13]

This is an extension to Performance Optimized Routing algorithm, It does not consider location of the resource based on SLA and if the best response time is not from the closest data center then it selects closest data center or data center with the least response time with the 50% probability. In this algorithm selection of the resource based on SLA is considered and when there is more than one resources are available it selects the data center with the least execution cost and also it manages the load between the datacenters.

## 2.1 Service Broker Algorithms

### 2.1.1 Service Proximity Based Routing[2]

This is the simplest Service Broker implementation.

1. ServiceProximityServiceBroker maintains an index table of all Data Centers indexed by their region.



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

2. When the Internet receives a message from a user base it queries the ServiceProximityServiceBroker for the destination DataCenterController.
3. The ServiceProximityServiceBroker retrieves the region of the sender of the request and queries for the region proximity list for that region from the InternetCharacteristics. This list orders the remaining regions in the order of lowest network latency first when calculated from the given region.
4. The ServiceProximityServiceBroker picks the first data center located at the earliest/highest region in the proximity list. If more than one data center is located in a region, one is selected randomly.

## 2.1.2 Performance Optimized Routing[2]

This policy is implemented by the BestResponseTimeServiceBroker, which extends the Service Proximity Service Broker.

1. Best Response Time Service Broker maintains an index of all Data Centers available.
2. When the Internet receives a message from a user base it queries the Best Response Time Service Broker for the destination Data Center Controller.
3. The Best Response Time Service Broker identifies the closest (in terms of latency) data center using the Service Proximity Service Broker algorithm.
4. Then the Best Response Time Service Broker iterates through the list of all data centers and estimates the current response time at each data center by
  - a. Querying the last recorded processing time from Internet Characteristics.
  - b. If this time is recorded before a predefined threshold, the processing time for that data center is reset to 0. This means the data center has been idle for a duration of at least the threshold time.
  - c. The network delay from Internet Characteristics is added to the value arrived at by above steps.
5. If the least estimated response time is for the closest data center, the Best Response Time Service Broker selects the closest data center. Else, Best Response Time Service Broker picks either the closest data center or the data center with the least response time with a 50:50 chance (i.e. load balanced 50:50).

## 2.1.3.Enhanced Proximity-Based Routing Algorithm [14]

1. ServiceProximityServiceBroker maintains an index table of all Data Centers indexed by their region.
2. When the Internet receives a message from a user base it queries the ServiceProximity ServiceBroker for the destination DataCenterController.
3. The ServiceProximityServiceBroker retrieves the region of the sender of the request and queries for the region proximity list for that region from the InternetCharacteristics.This list orders the remaining regions in the order of lowest network latency first when calculated from the given region.
4. The ServiceProximityServiceBroker picks the first data center located at the earliest/highest region in the proximity list and satisfied by SLA. If more than one data center is located in a region, select the datacenter from the region where  $Vm\$/Hr$  is minimum. When there are multiple datacenters with the same  $Vm\$/Hr$  is present , select the datacenter randomly.

## 2.1.4 . Resource Location and Cost based Performance Optimized Routing [13]

- i. BestResponseTimeServiceBroker maintains an index of all Data Centers based on the user SLA , by default it maintains the index of all Data Centers
- ii. When the Internet receives a message from a user base it queries the BestResponseTimeServiceBroker for the destination DataCenterController.
- iii. The BestResponseTimeServiceBroker identifies the closest (in terms of latency) data center using the ServiceProximityServiceBroker algorithm and selects data center that are listed in SLA
- iv. Then the BestResponseTimeServiceBroker iterates through the list of all data centers and estimates the current response time at each data center by
  - a. Querying the last recorded processing time from InternetCharacteristics.
  - b. If this time is recorded before a predefined threshold, the processing time for that data center is reset to 0. This means the data center has been idle for a duration of at least the threshold time.
  - c. The network delay from InternetCharacteristics is added to the value arrived at by above steps.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

v. If the least estimated response time is for the closest data center, the BestResponseTimeServiceBroker selects the closest data center. Else selects datacenter with the least response time.  
When there are multiple datacenters are available it selects the datacenter where the processing cost is minimum.

## III. SIMULATION RESULTS AND ANALYSIS

The proposed algorithm is simulated in a simulation toolkit CloudAnalyst[2]

### 3.1 Introduction to Cloud Analyst

To support the infrastructure and application-level requirements such as modelling of on demand virtualization arising from Cloud computing paradigm, enabled resource simulators are required. Few simulators like CloudSim [1] and CloudAnalyst [2] are available. CloudAnalyst has been used in our paper as a simulation tool. A snapshot of the GUI of CloudAnalyst simulation toolkit is shown in figure 1(a) and its architecture is shown figure 1(b).

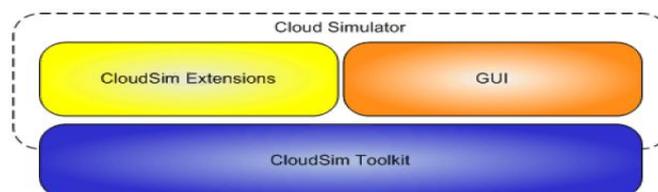


Figure 1. CloudAnalyst built on top of CloudSim toolkit



Fig 2Architecture of Cloud Analyst build on CloudSim

CloudAnalyst developed on CloudSim is a GUI based simulation tool. CloudSim facilitates modelling, simulation and other experimentation on cloud programmatically. CloudAnalyst uses the functionalities of CloudSim with GUI based simulation. It allows setting of parameters for setting a simulation environment to study any research problem of cloud. Based on the parameters the tool computes, the simulation result also shows them in graphical form. A hypothetical configuration has been generated using CloudAnalyst. Where, the world is divided into 6 „Regions” that coincide with the 6 main continents in theWorld. Six “User bases” modeling a group of users representing the six major continents of the world is considered. A single time zone has been considered for the all the user bases and it is assumed that there are varied number of online registered users during peak hours, out of which only one twentieth is online during the off-peak hours.

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

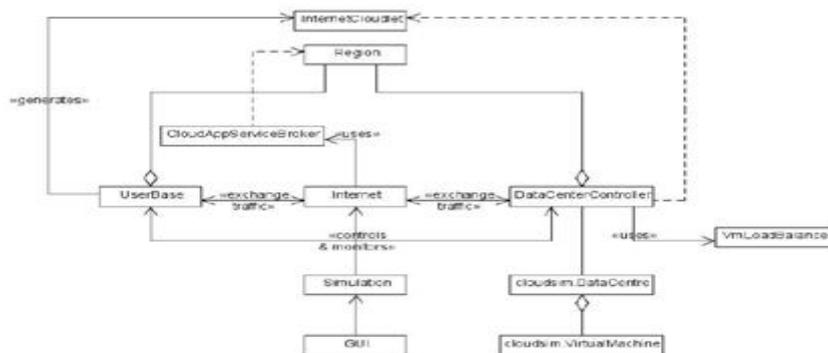


Figure 3 summarises the main components and domain entities of the CloudAnalyst.

The main components and their responsibilities are as follows:

- Gui – The main class of the GUI. Displays the GUI and acts as the front end controller for the application managing screen transitions and other UI activities.
- Simulation – Responsible for holding the simulation parameters and creating and executing the simulation.
- UserBase – Models a user base and generates traffic representing the users.
- DataCenterController – Controls the data center activities as explained above.
- Internet – Models the Internet and implements the traffic routing behaviour.
- InternetCharacteristics – Maintains the characteristics of the internet including the latencies and available bandwidths between regions, the current traffic levels and current performance level information for the data centers.
- CloudAppServiceBroker and its implementations – Models the service brokers
- VmLoadBalancer and its implementations – Models the load balancing logic used by data centers in allocating requests to VMs.
- UserBaseUIElement, DataCenterUIElement, MachineUIElement – holds information about user bases, data centers and machines for the UI until Simulation use them to create the respective simulation entities.

### 3.2 Simulation scenario1

In this scenario User Base and data center configurations are as given in Table 1 and Table 2 and it is assumed that UB1 user requests can be executed in any datacenter except in Region 1. In the Proximity-Based Routing algorithm always selects closest data center and most of the cases data center within the user base.

S.No	UserBase	Region	Onlineusers duringpeakhrs.	Onlineusersduringoff-peak
1	UB1	1	1000	100

Table 1 : Configuration of User Bases

Name	Region	Arch	OS	VMM	Cost per VM \$/Hr	Memory Cost \$/s	Storage Cost \$/s	Data Transfer Cost 4/Gb
DC1	1	x86	Linux	Xen	2	0.05	0.1	0.1
DC2	1	x86	Linux	Xen	1	0.05	0.1	0.1
DC3	2	x86	Linux	Xen	1	0.05	0.1	0.1
DC4	2	x86	Linux	Xen	1	0.05	0.1	0.1
DC5	4	x86	Linux	Xen	1	0.05	0.1	0.1

Table 2 : Configuration of Data Centers:

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

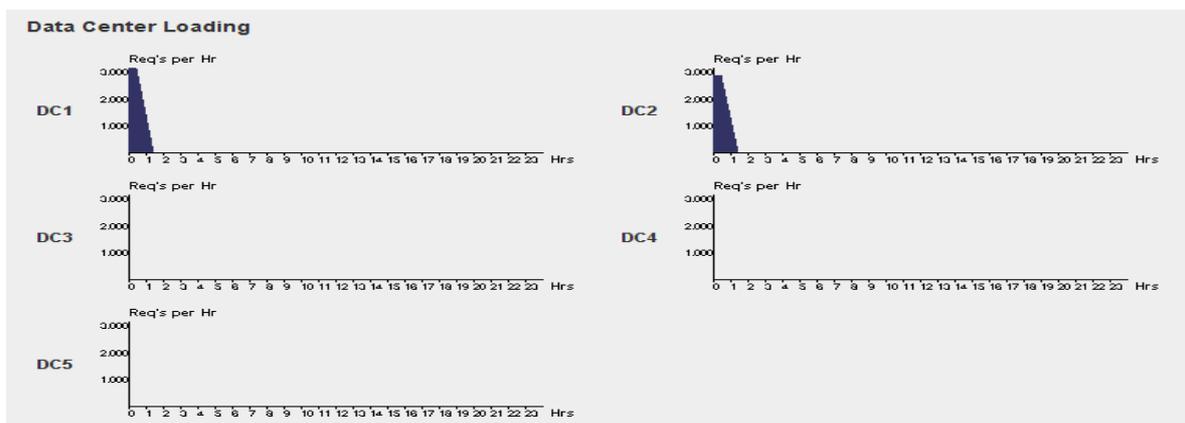


Fig 4:Datacenter Loading - Service Proximity based routing algorithm



Fig 5: Simulation result - Service Proximity based routing algorithm

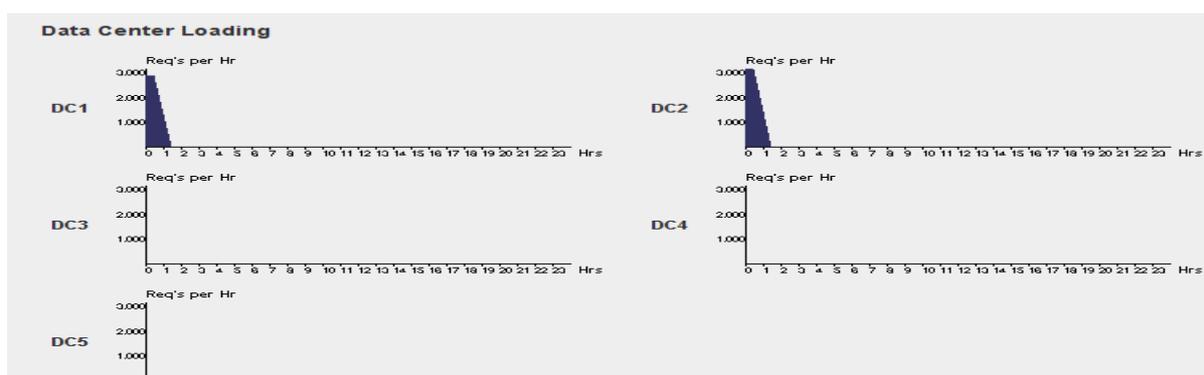


Fig 6:Datacenter Loading -Performance Optimized Routing

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

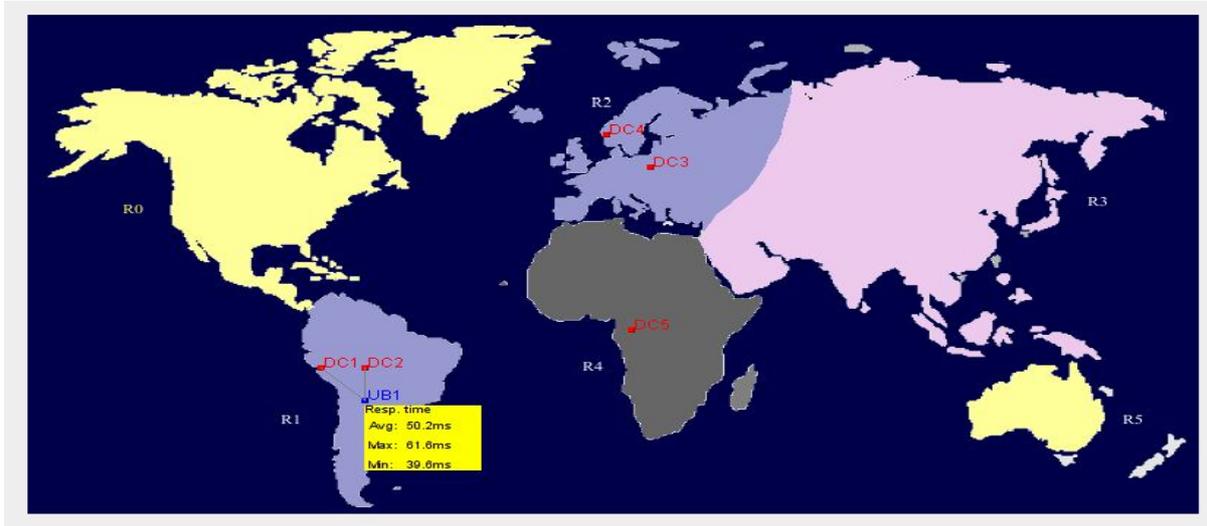


Fig 7: Simulation result -Performance Optimized Routing

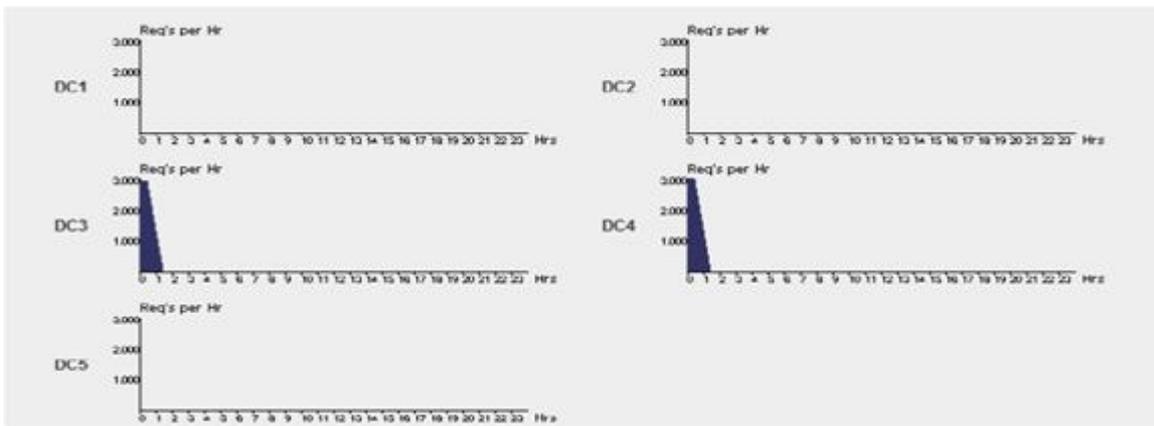


Fig 8:Datacenter Loading - Resource Location and Cost based Performance Optimized Routing



Fig 9: Simulation result - Resource Location and Cost based Performance Optimized Routing



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

Fig 4 and Fig 6 shows the results of datacenter loading .It shows the loads are equally distributed between the datacenters.. Here UB1 region is 1. Results shows that UB1 requests are processed in data centers DC1 and DC2 .The assignment of the user bases to the data centers also depicted in the final simulation results as shown in Fig 5 and fig 7. These experiments shows thatService Proximity based routing algorithm always it selects the closest datacenter and performance optimised algorithm selectsIf least response time data center is the closest data center then it selects the closest data center else selects closest data center and datacenter with the least response time with a 50:50 chance.

Fig 8 shows the results of data center loading . Here UB1 region is 1 and the SLA requirement is request can be executed in any region other than region1. Results shows that UB1 requests are processed in data centers DC3 and DC4 . It satisfies SLA requirement . The data centers DC3 and DC4 are in region 2 and cost per VM\$/Hr is same , the Fig 8 is also shows that load is equally distributed between the data centers. The assignment of the user bases to the data centers also depicted the final simulation results as shown in Fig 9.

### 3.2 Simulation scenario2

In this scenario User Base and data center configurations as given in Table 3 and Table 4 and It is assumed that UB1 user requests can be executed in any datacenter except in region 1 and region 3. UB2 user requests can be executed in any datacenter except in region 2 and 4 and UB3 can be executed in any datacenter.

S.No	User Base	Region	Online users during peak hrs.	Online users during off-peak
1	UB1	1	1000	100
2	UB2	2	1000	100
3	UB3	3	1000	100

Table 3 : Configuration of User Bases

Name	Region	Arch	OS	VMM	Cost per VM \$/Hr	Memory Cost \$/s	Storage Cost 4/s	Data Transfer Cost 4/Gb
DC1	1	x86	Linux	Xen	2	0.05	0.1	0.1
DC2	1	x86	Linux	Xen	1	0.05	0.1	0.1
DC3	2	x86	Linux	Xen	1	0.05	0.1	0.1
DC4	2	x86	Linux	Xen	1	0.05	0.1	0.1
DC5	4	x86	Linux	Xen	1	0.05	0.1	0.1

Table 4 : Configuration of Data Centers:

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

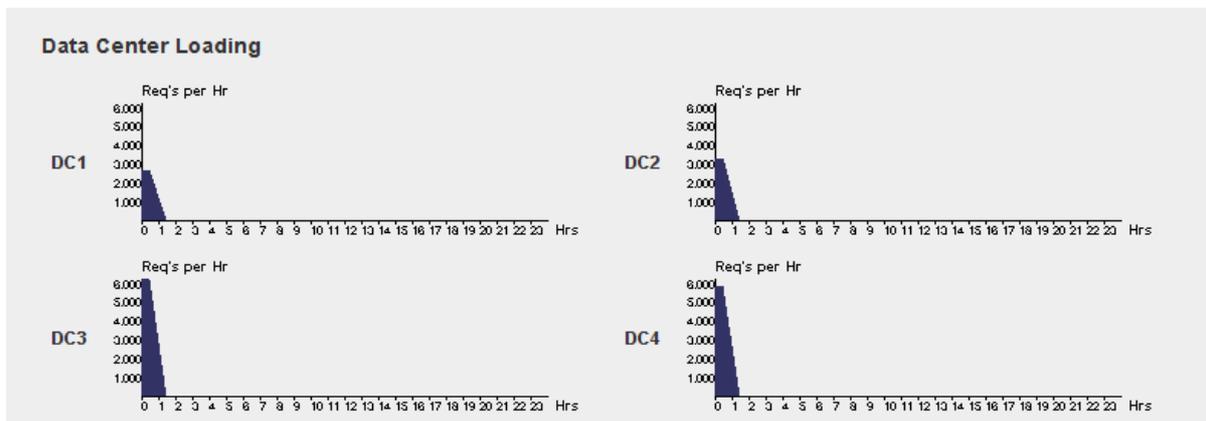


Fig 10:Datacenter Loading - Service Proximity based routing algorithm

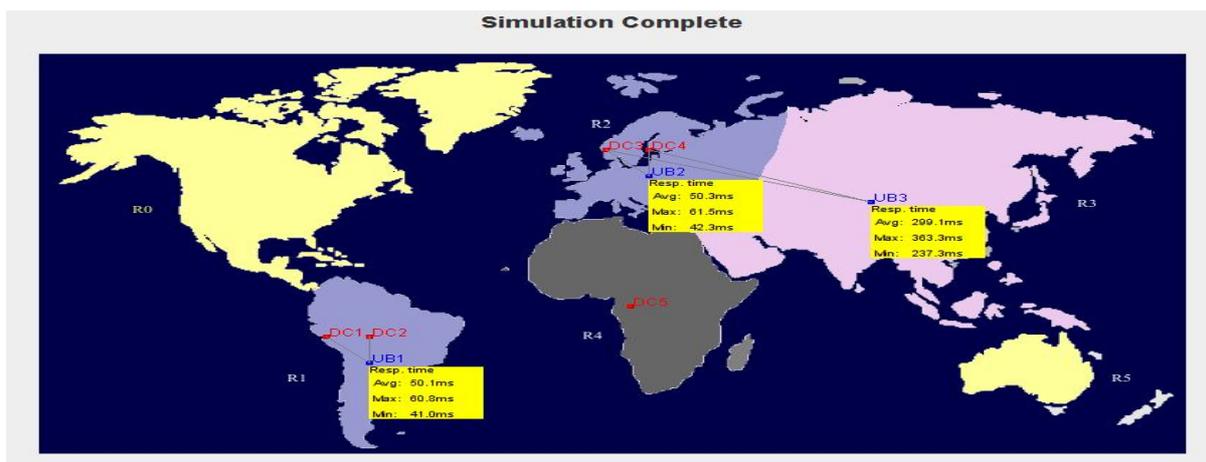


Fig 11: Simulation result - Service Proximity based routing algorithm

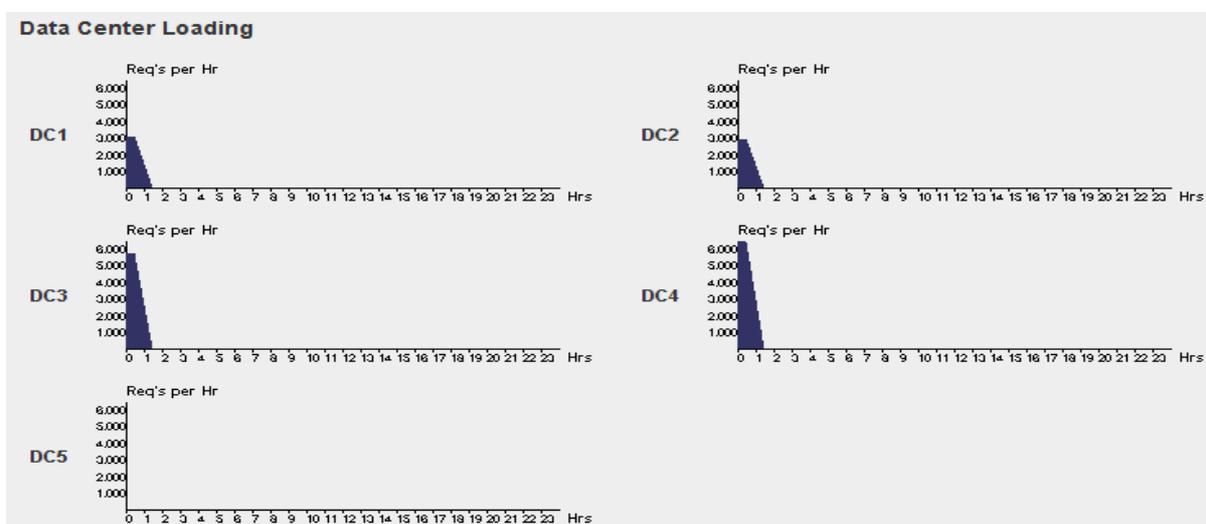


Fig 12 :Datacenter Loading -Performance Optimized Routing

# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014



Fig 13: Simulation result -Performance Optimized Routing

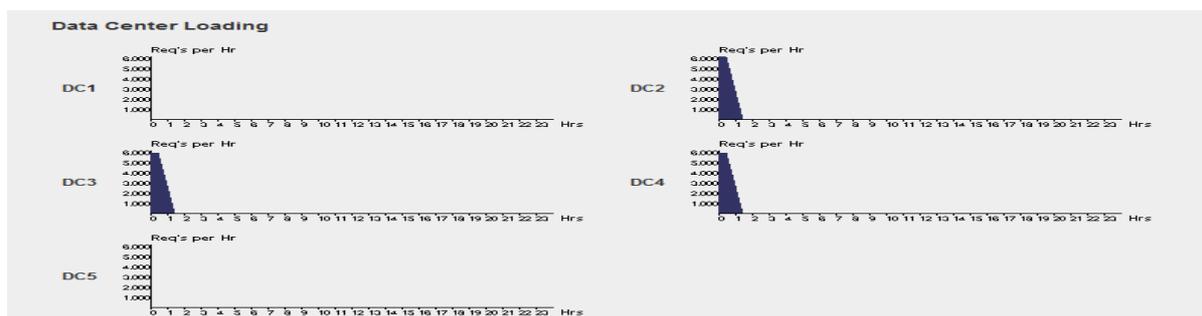


Fig 14: Datacenter Loading - Resource Location and Cost based Performance Optimized Routing

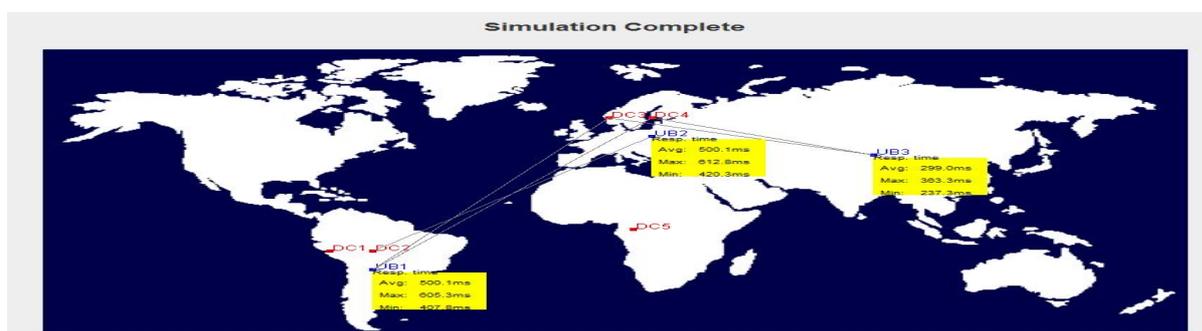


Fig 15: Simulation result - Resource Location and Cost based Performance Optimized Routing

Fig 10 and Fig 12 shows the results of datacenter loading .it shows the loads are equally distributed between the datacenter. The assignment of the user bases to the data centers also depicted in the final simulation results as shown in Fig 11 and fig 13. These experiments shows that Service Proximity based routing algorithm always it selects the closest datacenter and performance optimised algorithm selects If least response time data center is the closest datacenter then it selects the closest datacenter else selects closest data center and datacenter with the least response time with a 50:50 chance.

Fig 14 shows the results of data center loading .Here UB1,UB2,UB3 user bases are in the region 1,2 and 3 respectively. The SLA requirement for UB1 is request can be executed in any region other than 1 and 3, UB2 request can be



# International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol.2, Special Issue 5, October 2014

executed in any region other than 2 and 4 and there is no restriction of data centers for userbase UB3. The results in Fig 14 and Fig 15 shows that the UB1 region is 1 and closest are data centers are DC1 and DC2 , requests are executed in region 3 by the data centers DC3 and DC4 and satisfies the given SLA. The SLA requirement for UB2 is request can be executed in any region other than 2 and 4. The results in Fig 14 and Fig 15 shows that though UB2 region is 2 and closest are data centers are DC3 and DC4 , requests are executed in region 1 by the data center DC1 and DC2 . Here cost of VM\$/Hr of data center of DC1 is more than DC2. Therefore majority of the requests are executed in DC2. Since there is no restrictions on the user base UB3 , results shows that it was executed in the closest data centers DC3 and DC4. Here the cost of VM\$/Hr of data centers are same and the fig 14 shows that the load is equally distributed between the data centers. The assignment of the user bases to the data centers also depicted the final simulation results as shown in Fig 15

## IV. CONCLUSION

Scheduling is one of the most important task in cloud computing environment. In this paper we have explained different types of scheduling and analyzed various service broker algorithms and compared its data center loading and assignment of user base requests to datacenter. For assigning the requests the algorithm uses different parameters like closest path, performance and location of the resource etc.

## ACKNOWLEDGMENT

The authors would like to thank the dedicated research group in the area of Cloud & Grid Computing , wireless networks at the Dept of MCA ,AIMIT,St Aloysius College, Mangalore and Mangalore University, Mangalore, India, for many stimulating discussions for further improvement and future aspects of the Paper. Lastly but not least the author would like to thank everyone, including the anonymous reviewers.

## REFERENCES

- [1] R.N.Calheiros,R.Ranjan,A.Beloglazov,C.Rose,R.Buyya,“Cloudsim:A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms”,in *Software:Practice and xperience(SPE)*,Vol:41,No:1,ISSN:0038-0644,Wiley Press,USA,pp:23-50,2011.
- [2] B. Wickremasinghe, R.N. Calheiros and R. Buyya, “Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications”, in Proc. of *Proceedings of the 24th International Conference on Advanced Information Networking and Applications (AINA 2010)*, Perth, Australia, pp.446-452, 2010.
- [3] Chieu T.C., Mohindra A., Karve A.A., Segal A., “Dynamic Scaling of Web Applications in a Virtualized Cloud Computing Environment ,” in IEEE International Conference on e-Business Engineering, Dec. 2009, pp. 281-286.
- [4] Naidila Sadashiv, S. M Dilip Kumar, “Cluster, Grid and Cloud Computing: A Detailed Comparison,” The 6th International Conference on Computer Science & Education (ICCSE 2011) August 3-5, 2011. SuperStar Virgo, Singapore, pp. 477- 482.
- [5] Vincent C. Emeakaroha, Ivona Brandic, Michael Maurer, Ivan Breskovic, “SLA-Aware Application Deployment and Resource Allocation in Clouds”, 35th IEEE Annual Computer Software and Application Conference Workshops, 2011, pp. 298-303.
- [6] V. C. Emeakaroha, I. Brandic, M. Maurer, and S. Dustdar, “Low level metrics to high level SLAs - LoM2HiS framework: Bridging the gap between monitored metrics and SLA parameters in cloud environments,” In High Performance Computing and Simulation Conference, pages 48 – 55, Caen, France, 2010.
- [7] M. Maurer, I. Brandic, V. C. Emeakaroha, and S. Dustdar, “Towards knowledge management in self-adaptable clouds,” In 4th International Workshop of Software Engineering for Adaptive Service-Oriented Systems (SEASS’10), Miami, Florida, USA, 2010.
- [8][http://www.asd.gov.au/publications/csocprotect/Cloud\\_Computing\\_Security\\_Considerations.pdf](http://www.asd.gov.au/publications/csocprotect/Cloud_Computing_Security_Considerations.pdf)
- [9] R. Van den Bossche, K. Vanmechelen, J. Broeckhove, Cost-optimal scheduling in hybrid iaas clouds for deadline constrained workloads, in: IEEE 3rd International Conference on Cloud Computing CLOUD, pp. 228–235.
- [10] R. Buyya, C. Yeo, S. Venugopal, J. Broberg, I. Brandic, Cloud computing and emerging IT platforms: vision, hype, and reality for delivering computing as the 5th utility, *Future Generation Computer Systems* 25 (2009) 599–616.
- [11] Martin Randles, David Lamb, A. Taleb-Bendiab, “A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing”, 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops, pp. 551-556.
- [12] Soumya Ray and Ajanta De Sarkar, “Execution Analysis Of Load Balancing Algorithms In Cloud Computing Environment”, *International Journal on Cloud Computing: Services and Architecture (IJCCSA)*, Vol.2, No.5, October 2012
- [13] SanthoshB ,Dr D.H Manjaiah “Resource Location and Cost based Performance Optimized Routing algorithm in cloud computing”Second International Conference on Emerging Research in Computing, Information, Communication and Applications, August 2014 published in ELSEVIER proceedings with ISBN 9789351072621
- [14] SanthoshB ,Dr D.H Manjaiah “Enhanced Proximity-Based Routinng Policy for ServiceBrokering in Cloud Environment”,National Conference on Recent Trends in Computing &Information Technology, August 2014 ISBN N.978-81-929306-2-6