Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 5, May 2014, pg.444 – 453

RESEARCH ARTICLE

Efficient Virtual Machine Scheduling in Cloud Computing

Dilshad H.Khan, Prof. Deepak Kapgate
Department of C.S.E., GHRAET, Nagpur, Nagpur University
Khan27jan@gmail.com, deepakkapgate32@gmail.com

Abstract - Due to rapid increase in use of Cloud Computing, moving of more and more applications on cloud and demand of clients for more services and better results, load balancing in Cloud has become a very interesting and important research area. VM Scheduling is essential for efficient operations in distributed environments. In cloud computing the load balancing concept broadly classify in three stages as Data Centre Selection, Virtual Machine Scheduling and Task scheduling at particular data centre. Many algorithms were suggested to provide efficient mechanisms and algorithms for assigning the client's requests to available Cloud nodes. In this paper, we explained different algorithms and techniques proposed for Virtual Machine Scheduling either at single data centre or multiple data center. Also infers their characteristics to resolve the issue of efficient Virtual Machine Management in Cloud Computing. We discuss and compare these algorithms and techniques in regards of various performance matrices to provide an overview of the latest approaches in the field.

Keywords— Cloud Computing, VM management, scheduling algorithms, data center, scheduling techniques

I. INTRODUCTION

Any solution where data storage and any processing take place without the user being able to pinpoint the specific computer carrying. Cloud computing refers to both thee application delivered as services over the internet and the hardware and system software in the data center that provides those services. Cloud computing provides shared pool of resources on-demand over network on pay per use. Cloud computing insures access to virtualized it resources that data center are presented and are shared by others. It is common to divide cloud computing into three categories:

A. Infrastructure as a service (IaaS)

It provides flexible ways to create use and manage virtual machines. In IaaS model, computing resources such as storage, network, and computation resources are provisioned as services. Consumers are able to deploy and run arbitrary software, which can include operating systems and applications. Consumers do not manage or control the underlying cloud infrastructure but have to control its own virtual infrastructure typically constructed by virtual machines hosted by the IaaS vendor. This thesis work mainly focuses on this model, although it may be generalized to also apply to the other models.

B. Platform as a service (PaaS)

Focused on providing the higher level capabilities more than just virtual machines required to supports applications. In the PaaS model, cloud providers deliver a computing platform and/or Solution stacks typically including operating system, programming language execution environment, database, and web server [5]. Application developers can develop and run their software on a cloud platform without having to manage or control the underlying hardware and software layers, including network, servers, operating systems, or storage, but maintains the control over the deployed applications and possibly configuration settings for the application-hosting environment.

C. Software as a service (SaaS)

The application that provides business value for users. In the SaaS model, software applications are delivered as services that execute on infrastructure managed by the SaaS vendor. Consumers are enabled to access services over various clients such as web browsers and programming interfaces, and are typically charged on a subscription basis [6]. The implementation and the underlying cloud infrastructure where it is hosted is transparent to consumers.

D. Deployment Models

The cloud computing deployment model describes where the software runs and includes the following options: Based on the classification of cloud services into SaaS, PaaS, and IaaS, two main stakeholders in a cloud provisioning scenario can be identified, i.e., the Infrastructure Provider (IP) who offers infrastructure resources such as Virtual Machines, networks, storage, etc. which can be used by Service Providers (SPs) to deliver end-user services such as SaaS to their consumers, these services potentially being developed using PaaS tools. As identified in [7], four main types of cloud scenarios can be listed as follows.

1) *Private cloud:* Private cloud is set of standardized computing resources that is dedicated to an organization , usually on-premises in the organization data centers .it works with the current capital investment and drivers the new function as a service.

2) *Cloud Bursting:* Private clouds may offload capacity to other IPs under periods of high workload, or for other reasons, e.g., planned maintenance of the internal servers.

3) *Federated Cloud:* Federated Cloud are cloud collaborated on a basis of load sharing agreements enabling them to offload capacity to each other's in a manner similar to how electricity providers exchange capacity.

4) *Multiple clouds:* In multi-cloud scenarios, the SP is responsible for handling the additional complexity of coordinating the service across multiple external IPs, i.e. planning, initiating and monitoring the execution of services.

E. Parameters of interest for cloud services Provider

1) *Resources utilization details:* Just like any other performance monitoring utilization parameter of physical server infrastructure is an important factor in cloud monitoring, as these service make up the cloud.

2) *Infrastructure response time (IRT):* IRT gives the clear picture of the overall performance of the cloud as it checks the time taken for each transaction to complete.

3) *Virtualization metrics:* Similar to the physical machine, we need to collect the resource utilization data from the virtual machines. This provides the picture of how much of the virtual machine is being utilized and this data helps in the resources utilization by application and divided on the scale requirements.

4) Transaction matrices: It can be considered as derivative from IRT . Metrics like success percentage of transaction counts of transaction etc. for an application would give a clear picture of the performance of an application in cloud particular instant.

Cloud computing enjoys the many attractive attributes of virtualization technology, such as consolidation, isolation, migration and suspend/resume support. A virtual machine (VM) is a software implementation of a computing environment in which an operating system (OS) or program can be installed and run. Important parameters related to virtual machines are Number of virtual machines used by applications, Time taken to create a new VM, Time taken to move an application from one VM to another, Time taken to allocate additional resources to VM. Virtualization is the creation of a virtual version of something such as an operating system, a server, a storage device or network resources.

Scheduling the basic processing units on a computing environment has always been an important issue [1]. Like any other processing unit, VMs need to be scheduled on the cloud in order to Maximize utilization, Do the job faster, Consume less energy, Easy resource reservation (allocation). VM's elasticity in cloud computing, elasticity is defined as the degree to which a system is able to work loud change by provisioning and de-provisioning resources in an automatic manner such that at each point in time the available resources match the current demand as closely as possible.

The number of cloud users has been growing exponentially and apparently scheduling of virtual machines. In the cloud becomes an important issue to analyze. In cloud computing, a user may require a set of virtual machine co-operating with each other to accomplish one task. In the past the inter relationship among task are not considered. Scheduling is the method by which virtual machine flows are given access to system resources.

Individual virtual machine throughput, but also on the activation latency and response-time by which virtualized software components react to external events. A real system validating the approach by recurring to soft real-time scheduling strategies at the virtualization layer, it is possible to provide a good level of isolation between the concurrently running VMs. Furthermore, it is possible to achieve both a good throughput of the VMs and to keep the individual guarantees at the latency level, something that is not possible with the standard Linux scheduling strategies.

II. Literature Survey

The goal of scheduling algorithms in distributed systems is spreading the load on processors and maximizing their utilization while minimizing the total task execution time Job scheduling, one of the most famous optimization problems, plays a key role to improve flexible and reliable systems. The main purpose is to schedule jobs to the adaptable resources in accordance with adaptable time, which involves finding out a proper sequence in which jobs can be executed under transaction logic constraints [2].

C. Reddy [7] explain use of gang scheduling algorithm in cloud computing responsible for selection of best suitable resources for task execution, by taking some static and dynamic parameters and restrictions of VM into the considerations. Gang scheduling is a scheduling algorithm for parallel system that scheduled related VM to run simultaneously on different machines. Gang Scheduling is an efficient job scheduling algorithm for time sharing, already applied in parallel and distributed systems. Gang scheduling can be effectively applied in a Cloud Computing environment both performance-wise and cost-wise. Gang scheduling is a special case of job scheduling that allows the scheduling of such virtual Machines.

Round Robin is proportionally fair algorithm, or maximum throughput scheduling (throughput). The main advantage of this algorithm is that it utilizes all the resources in a balanced order (resource utilization). The scheduler starts with a node and moves on to the next node, after a VM is assigned to that node. This is repeated until all the nodes have been allocated at least one VM and then the scheduler returns to the first node again. Hence, in this case, the scheduler does not wait for the exhaustion of the resources of a node before moving on to the next (Fault tolerant) [6].

Content-Based Virtual Machine Scheduling Algorithm - The content based VM scheduling algorithms were designed with the goal of lowering the amount of data transferred between racks in the data center when virtual machines disk image are being copied to the host node [5]. The algorithm returns the selected node and the VM on that node with the highest similar content. When deploying a VM, we search for potential hosts that have VMs that are similar in content to the VM being scheduled. Then, we select the host that has the VM with the highest number of disk blocks that are identical to ones in the VM being scheduled.

III. Proposed Improved Load Balance Min Min Algorithm

Improved Load Balance Min Min Algorithm (ILBMM):

- ILBMM is presented in Figure 1. The algorithm starts by executing the steps in Min-Min strategy first.
- It first identifies the VM having minimum execution time and the resource producing it. Thus the VM with minimum execution time is scheduled first in Min-Min. After that it considers the minimum completion time since some resources are scheduled with some VM.
- So LBMM executes Min-Min in the first round. In the second round it chooses the resources with heavy load and reassigns them to the resources with light load.
- The completion time for that task is calculated for all resources in the current schedule. Then the maximum completion time of that VM is compared with the makespan produced by Min-Min.
- if it is less than makespan then the task is rescheduled in the resource that produces it, and the ready time of both resources are updated.
- Otherwise the next maximum completion time of that task is selected and the steps are repeated again. The process stops if all resources and all VM assigned in them have been considered for rescheduling.
- Thus the possible resources are rescheduled in the resources which are idle or have minimum load.

i) Request Manager :

  - Accept Request From User Base
  - Send Requests to Service Manager

ii) Service Manager:

- Count VMState list
- Counting of VM capability: verify remaining memory, CPU, space and bandwidth and also provides current allocation and sends Request to min.

Flowchart:

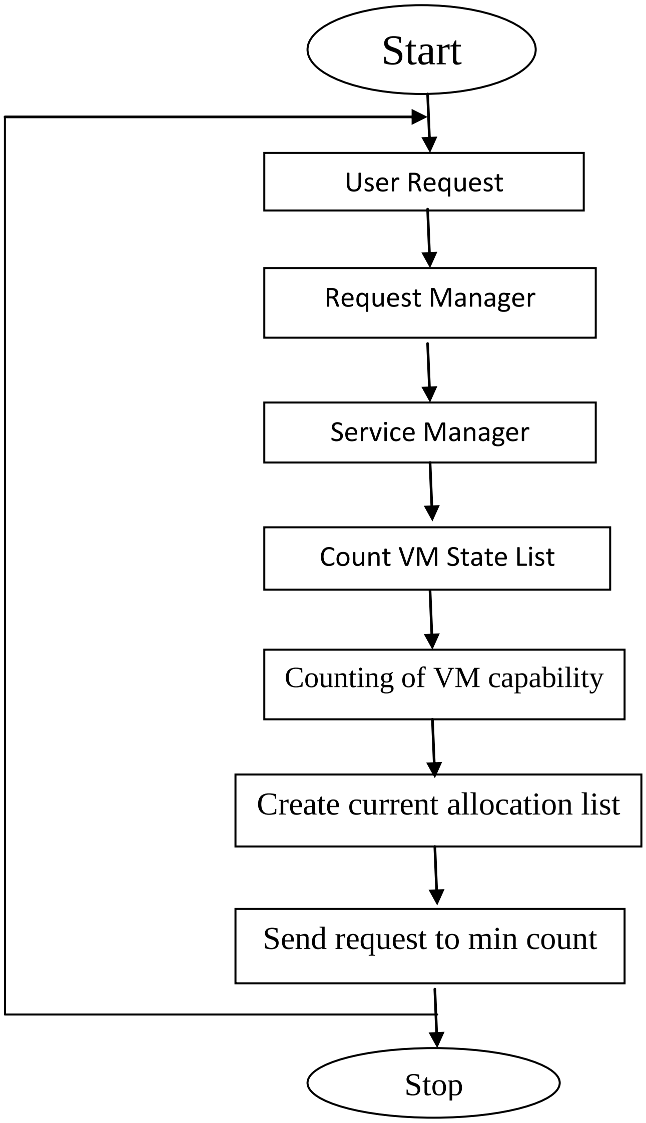

Start
User Request
Request Manager
Service Manager
Count VM State List
Counting of VM capability
Create current allocation list
Send request to min count
Stop

Figure1. ILBMM

ILBMM Algorithm:

Step-1 execute ILBMM (there frames)

Step-2 Request Manager: Accept request from user base and send request to service manager

Step-3 Service Manager: Count VM State List.

Step-4 counting of VM capability: verify remaining memory, CPU, space and bandwidth. and also provides current allocation and sends Request to min.

Step-5 if it is reasonable then stop otherwise repeat step-2.

IV. Implementation Details

The working environment for cloud computing where the proposed algorithm is implemented is done using cloud analyst simulator which is built above “CloudSim”, “GridSim” and “SimJava”. Cloud-Analyst is built on the top of Cloud-sim. Cloud-sim is developed on the top of the Grid-sim.

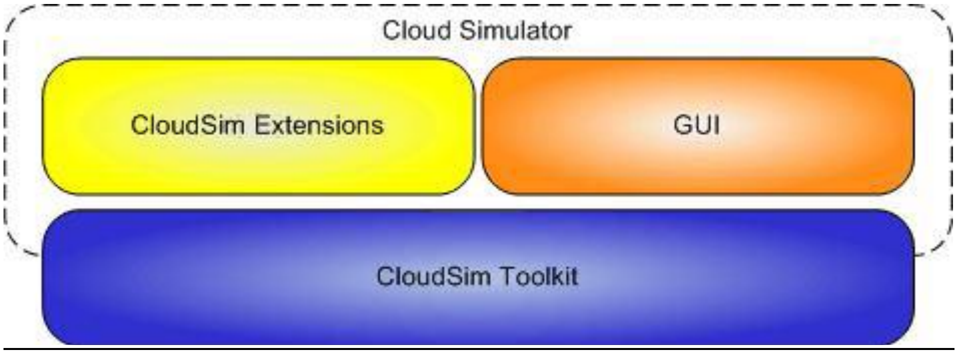


Figure2. Cloud-Analyst built on top of Cloud-Sim toolkit

- Application users - There is the requirement of autonomous entities to act as traffic generators and behavior needs to be configurable.
- Internet - It is introduced to model the realistically data transmission across Internet with network delays and bandwidth restrictions.
- Simulation defined by time period - In Cloud-sim, the process takes place based on the pre-defined events.Here, in Cloud-Analyst, there is a need to generate events until the set time-period expires.
- Service Brokers - DataCeneterBroker in CloudSim performs VM management in multiple data centers and routing traffic to appropriate data centers. These two main responsibilities were segregated and assigned to DataCenterController and CloudAppServiceBroker in Cloud-Analyst.

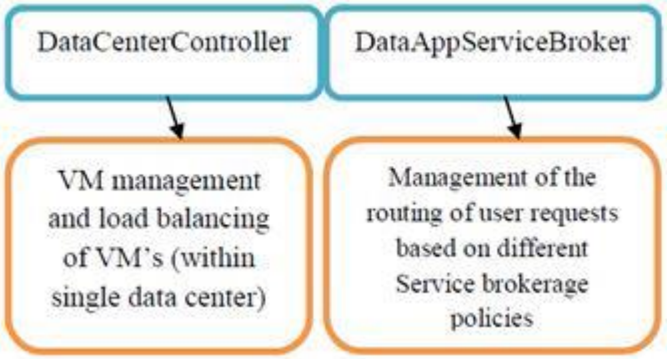
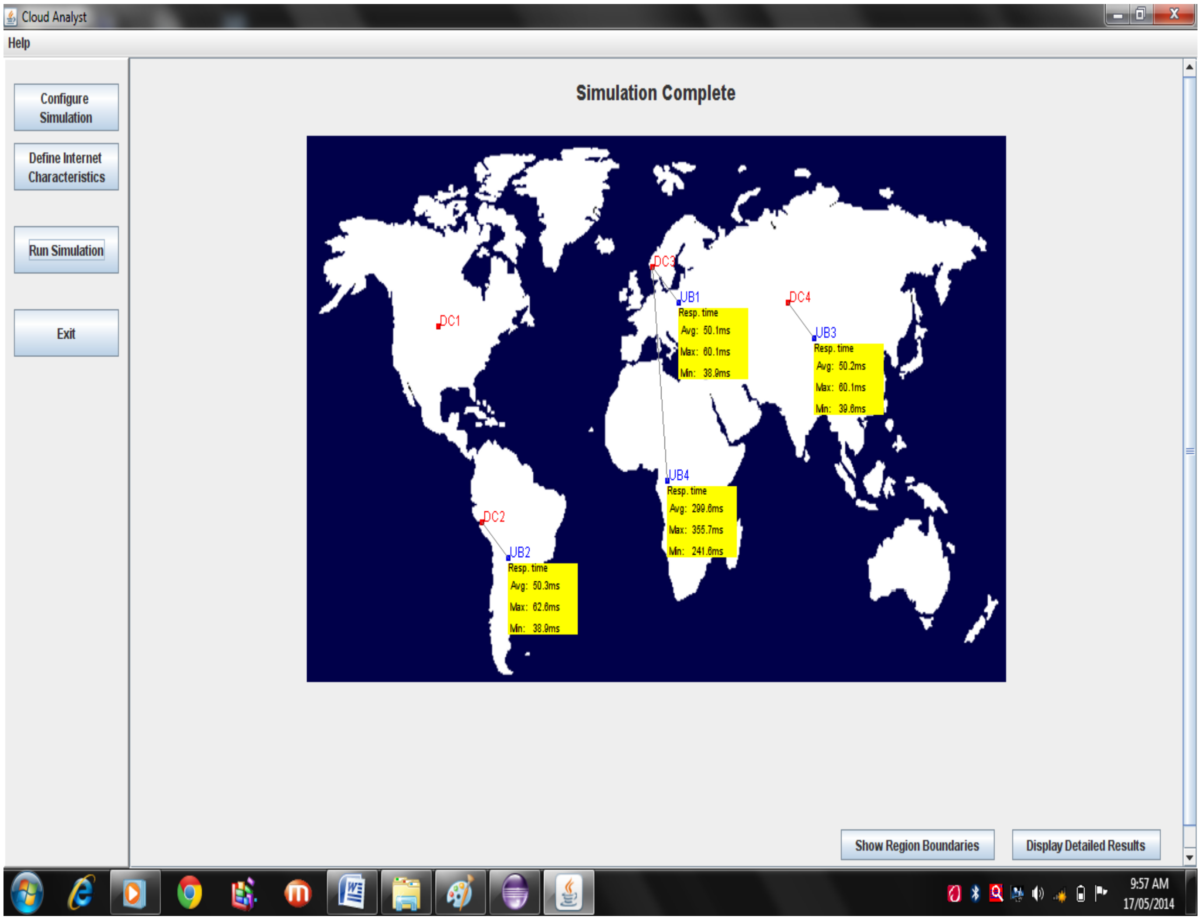

DataCenterController
DataAppServiceBroker
VM management and load balancing of VM's (within single data center)
Management of the routing of user requests based on different Service brokerage policies

Figure3. Responsibilities- Segregation

V. Results Calculated

The Proposed algorithm is implemented using simulation Cloud-Analyst. The scenario is taken where the data centers are located at different regions with user bases requesting services from different regions or from same region the final result screen shown below as –



UB1
Resp. time
Avg: 50.1ms
Max: 60.1ms
Min: 38.9ms

UB3
Resp. time
Avg: 50.2ms
Max: 60.1ms
Min: 39.6ms

UB4
Resp. time
Avg: 299.6ms
Max: 355.7ms
Min: 241.6ms

UB2
Resp. time
Avg: 50.3ms
Max: 62.6ms
Min: 38.9ms

Figure 4. Cloud Analyst Main Result Screen

In above screen fig. 3 the lines show that the user base is requesting service from corresponding data center or server with the values shown at boxes at each user bases represents the response time observed by respected user base. The values are the minimum response time calculated at client side wile requesting service from data center in the duration of simulation was running ,similarly it shows the maximum response time and the average response time from above two calculated values.

The Results calculated as values of Response time observed at each client side, Data Center Service Request Times, Total VM and Data Transfer Cost. These are shown as below:

VI. Comparative Analysis

A. Experiment 1 – Comparison of Response time observed by user.

The graph shows drastic reduction in average response timing observed by user for proposed algorithm as Improved VM Improved load balace min min forecast model.

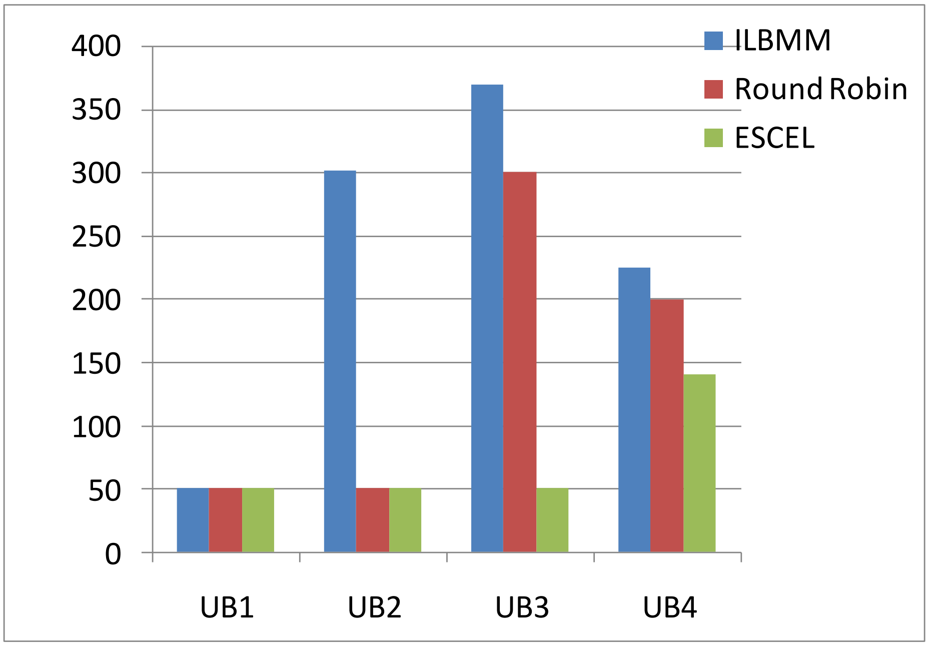


Figure 5. Response Time observed by user

B. Experiment 2 – Comparison of DC Request Service Times.

The graph shows increase in average DC Request Service Times for proposed algorithm as compared to traditional round robin service broker algorithm.

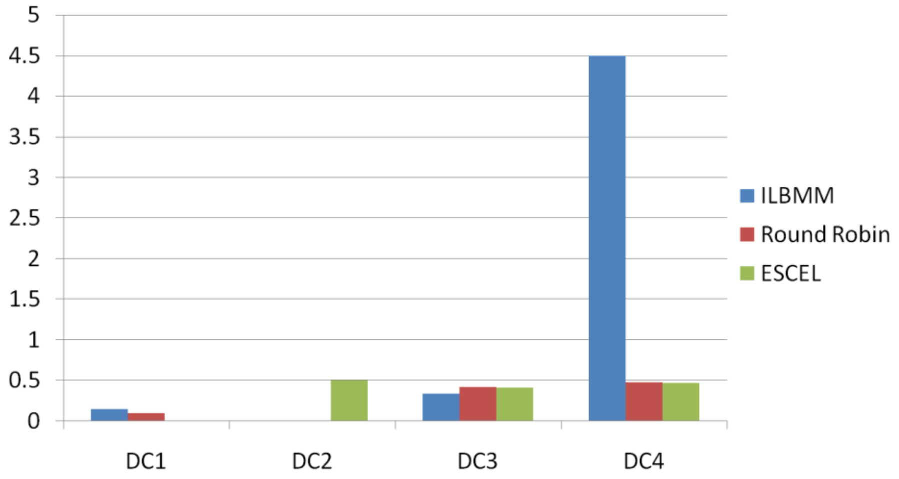


Figure6. Data Center Request Servicing Timing

C. Experiment 3 – Cost:

The graph shows Total cost of VM and Total data transfer cost.

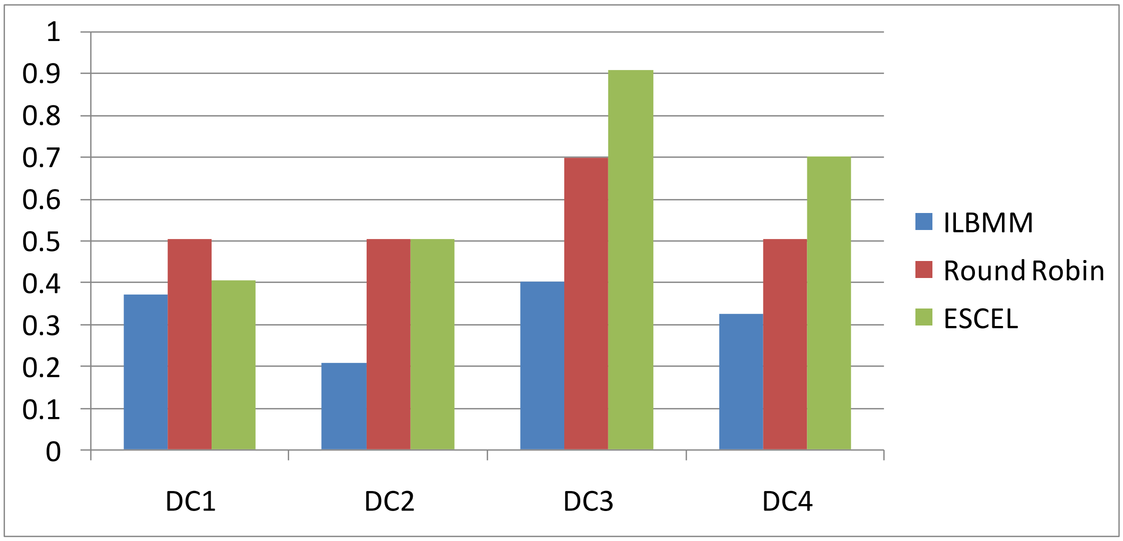


Figure7. Cost of VM

VI. CONCLUSION

The Proposed algorithm is implemented using simulation Cloud Analyst. The Scenario is taken where the data centers are located at different regions with user's bases requesting VM from different regions or from a same region. Here proposed algorithm based on Improved load balance min min (ILBMM) which drastically minimizes the response time observed by user which leads to improvement of service request timing is proposed.

REFERENCES

[1] Hadi Salimi , "Advantages, Challenges and Optimizations of Virtual Machine Scheduling in Cloud Computing Environments" in International Journal of Computer Theory and Engineering Vol. 4, No. 2, April 2012.

[2] Pinal Salot , "A Survey Of Various Scheduling Algorithm In Cloud Computing Environment" in M.E, Computer Engineering, Alpha College of Engineering, Gujarat, India , Volume: 2 Issue: 2.

[3] MR.NISHANT, "Pre-Emptable Shortest Job Next Scheduling In Private Cloud Computing" in journal of information, knowledge and research computer engineering, NOV 12 TO OCT 13 | VOLUME – 02, ISSUE – 02.

[4] TARUN GOYAL, "Host Scheduling Algorithm Using Genetic Algorithm In Cloud Computing Environment", International Journal of Research in Engineering & Technology (IJRET) Vol. 1, Issue 1, June 2013.

[5] Sobir Bazarbayev," Content-Based Scheduling of Virtual Machines (VMs) in the Cloud" in University of Illinois at Urbana-Champaign, AT&T Labs Research.

[6] Kiran Kumar et. al., "An Adaptive Algorithm For Dynamic Priority Based Virtual Machine Scheduling In Cloud" in IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 6, No 2, November 2012.

[7] Dr. Chenna Reddy , "An Efficient Profit-based Job Scheduling Strategy for Service Providers in Cloud Computing Systems" in International Journal of Application or Innovation in Engineering & Management (IJAIEM) , Volume 2, Issue 1, January 2013.

[8] I. Moschakis, H. Karatza, "Performance and Cost evaluation of Gang Scheduling in a Cloud Computing System with Job Migrations and Starvation Handling" in Department of Informatics Aristotle University of Thessaloniki, Greece , IEE 2013.

[9] Ramkumar N, Nivethitha , "Efficient Resource Utilization Algorithm (ERUA) for Service Request Scheduling in Cloud" in International Journal of Engineering and Technology (IJET) , Vol 5 No 2 Apr-May 2013.

[10] K. Parrott et. al. ," Deadline Aware Virtual Machine Scheduler for Grid and Cloud Computing" in 2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops.

[11] Dimitris Hatzopoulos, "Dynamic Virtual Machine Allocation in Cloud Server Facility Systems with Renewable Energy Sources" at IEEE International Conference on Communications (ICC) 2013, Budapest, Hungary.

[12] Getzi Jeba Leelipushpam.P, "Live Virtual Machine Migration Techniques – A Survey" in International Journal of Engineering Research & Technology (IJERT), Vol. 1 , September – 2012.

[13] Vignesh V, Sendhil Kumar KS, Jaisankar N , " Resource management and scheduling in cloud environment" in International Journal of Scientific and Research Publications, Volume 3, Issue 6, June 2013.

[14] K. Rasmi and V. Vive , "Resource Management Techniques in Cloud Environment - A Brief Survey" in International Journal of Innovation and Applied Studies,Vol. 2 No. 4 Apr. 2013.

[15] Jeongseob Ahn, Changdae Kim, Jaeung Han," Dynamic Virtual Machine Scheduling in Clouds for Architectural Shared Resources".

[16] Tommaso Cucinotta," Providing Performance Guarantees to Virtual Machines using Real-Time Scheduling" in Tommaso Cucinotta, Dhaval Giani, Dario Faggioli, and Fabio Checconi Scuola Superiore Sant'Anna, Pisa, Italy.

[17] Junliang Chen,Bing Bing Zhou, "Throughput Enhancement through Selective Time Sharing and Dynamic Grouping" in 2013 IEEE 27th International Symposium on Parallel and Distributed Processing.

[18] Ioannis A. Moschakis," Evaluation of gang scheduling performance and cost in a cloud computing system" in Journal of Supercomputing, Volume 59 Issue 2, February 2012.

[19] Manoranjan Dash , " Cost Effective Selection of Data Center in Cloud Environment" in ISSN, Volume-2, Issue-1, 20131.

[20] Abirami S.P., Shalini Ramanathan (2012), "Linear Scheduling Strategy for Resource allocation in Cloud Environment", International Journal on Cloud Computing and Architecture, vol.2, No.1, February.

[21] Shikharesh Mujumdar (2011), "Resource management on cloud: Handling uncertainities in parameters and policies", CSI communications, edn. pp. 16-19.

[22] Nilabja Roy, Abhishek Dubey and Aniruddha Gokhale, "Efficient Autoscaling in the Cloud using Predictive Models for Workload Forecasting".

[23] Soramichi Akiyama, Takahiro Hirofuchi, Ryousei Takano, Shinichi Honiden (2012), "MiyakoDori: A Memory Reusing Mechanism for Dynamic VM Consolidation", Fifth International Conference on Cloud Computing, IEEE 2012.

[24] V. Venkatesa Kumar et. al , "Job Scheduling Using Fuzzy Neural Network Algorithm in Cloud Environment", International Journal of Man Machine Interface, Vol. 2, No. 1, March 2012.