Balanced Resource Utilization based Cloud Hosting Along With Server Consolidation

Mahendra Raghuvanshi
Dept. of information technology
SATI Vidisha, India (464001)

Vivek Sharma
Dept. of information technology
SATI Vidisha, India (464001)

ABSTRACT
In the cloud surroundings, resources (computing and data) stored in the datacenter are accessed on-demand by number of customers jointly. So there should be a mechanism that Maximize system performance, consumption of remaining resources (optimization) and minimize the resource leak, energy consumption (Server consolidation), but in these cloud computing domain resources are extremely dynamic and holistic in nature. By cause of this nature, full utilization of the resources is very difficult without the suited resource balancing. To improve the overall system performance, resources must be properly allocated; Load uniformly distributed on physical machines and the proper virtual machine (VM) allocation method must be used. Various virtual machines (VM) allocation method have been proposed for reducing the response time, resource handling and balancing of load in a datacenter environment but they are not efficient to minimizing the Energy Consumption and Server consolidation This paper includes some traditional VM scheduling techniques with their inconsistency.

Keywords
Scheduling, Load Balancing, Virtualization, cloud datacenter.

1. INTRODUCTION
It is a way of computing where extremely scalable IT-related computing powers are delivered "as a service" using Internet technologies to several clients. The term cloud computing is a comparatively new one, gaining prominence in 2008 as a means of describing Internet-based, distributed, parallel computing and its associated applications [1]. So different expert defined it in different ways, Cloud computing, a framework for enabling acceptable, and on-demand network access to a common group of computing resources [2], and is emerging as a new paradigm of large-scale distributed computing. It is a collection of distributed and parallel computing, which is a pool of virtualized and inter-connected resources that can be dynamically provided one or more unified computing resources based on SLA and the resources are provided as services by the vendor to the user on the basis of pay as you go. Figure 1 shows how user connects to the cloud and accesses the resources (i.e. application software and computing services) on demand. Cloud computing is also titled utility computing in which resources are accessed at anytime, anywhere and anyplace as a service as pay as you go basis via internet [3].

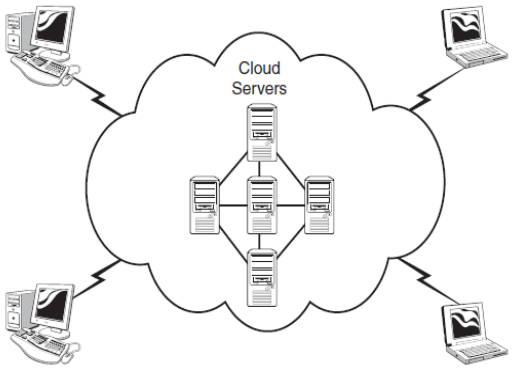

Figure 1 user interact to the cloud [1]

Cloud computing mainly deals with on application software, data storage services, and online computation, which may not require end-user knowledge of the physical locality and the composition of the system that is providing the services [4]. The figure 1.2 shows the architecture behind the cloud computing system, in which user only connects to the cloud The user's requests to the system management using front end interface, which calls the appropriate provisioning services. These services reserves requested resources in the cloud and launch the requested application. [17] Furthermore, system monitoring programs start determine uses of cloud resources for the proper user.

In a cloud computing domain VM allocation plays a critical act because cloud computing is a pool of composite resources that are allocated to different areas, so there is need to perfectly assign the consumer request to an appropriate physical resource with minimum overhead time and maximum resource utilization. Numbers of VM scheduling technology have been proposed in which different parameters are considered. VM scheduling can also be restricted on the basis of resources requirement, i.e. static and dynamic. By front end interface, how the user's requests are processed.

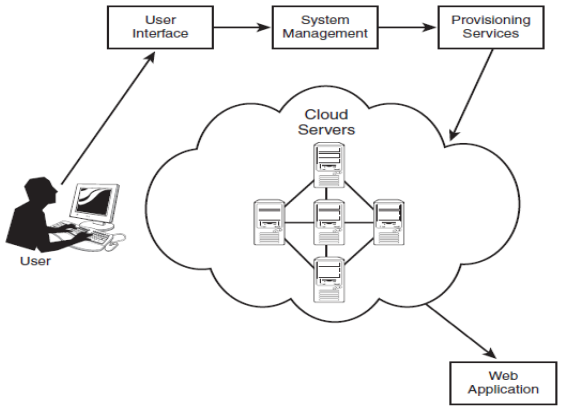

Figure 2 System Architecture of cloud [1].

In this paper, we are consulting with some advance VM allocation methods and their inconsistency based on the parameters, distribution of load, response time, resource usages, server consolidation and balancing of load.

2. RELATED WORK

A variety of virtual machine (VM) allocation techniques has been anticipated in the literature for maximizes system performance, utilization of available resources (optimization) and minimize the resource leak, energy usages (Server consolidation). W. Tian et al. [6] bring in an algorithm call DAIRS for allocating resource in datacenters, which generate a mean imbalance level of every physical machine and combined measurements for the entire imbalance impact of a cloud datacenter. X. Li et al. [7] suggested a balanced algorithm to decrease the energy usages by providing balance consumption of resources and preventing resource leaks. Subramanian S et al. [8] suggested a VM allocation method Greedy First Fit in this VM is allocated on the first running physical machine (PM) which satisfies the VM requirements. Due to the First Fit strategy of Greedy algorithm, there would be low resource utilization of available resources. S. K. Mandal et al. [9] planned an efficient VM placement algorithm based on binary search for access infrastructure resources on the basis of requirement in which cloud computing reduces the VM allocation time and resource usages in optimize way. Rajkumar Buyya et al. [10] projected a Modification Best Fit Decreasing (MBFD) technique which allocated resource in energy-efficient manner to provide server consolidation these are based on FFD (First Fit Decreasing technique) and also use the concept of lower and upper threshold.

3. LEAK OF RESOURCE OR IMBALANCE IMPECT MEASUREMENT

Xin Li et al. [7] gave the numerical explanation of resource leak. If there exists such R that (□□> □□) and (| max{μR} − min{μR}| ≥ θΔ), where □□ mention to the resource utilization of PM, □□ and □□ are the two threshold to find out the instant when a resource leak occurs. In this situation, it is pretended that all the PM are of the equal ability, but practically, in cloud environment the Physical machine can be of dissimilar capacities so in this resource leak or imbalance level is explained in different way. Let Bandwidth, Memory and CPU resource capacity of a physical machine (PM) are represented as Cpu, Mem and Bw respectively. Assume that Cpumax, Memmax, and Bwmax are the maximum available resource capacity of a physical machine (PM) and Cpurem, Memrem, and Bwrem are the remaining or available capacity of each resource. Remaining or available capacity of each resource in term of percentage is calculated as follows

$$\boldsymbol{Cpu} = \frac{\mathbf{Cpu_{rem}}}{\mathbf{Cpu_{max}}} * \mathbf{100} \quad (\mathbf{1})$$

$$\boldsymbol{Mem} = \frac{\mathbf{Mem_{rem}}}{\mathbf{Mem_{max}}} * \mathbf{100} \quad (\mathbf{2})$$

$$\boldsymbol{Bw} = \frac{\mathbf{Bw_{rem}}}{\mathbf{Bw_{max}}} * \mathbf{100} \quad (\mathbf{3})$$

$$\boldsymbol{Avg} = \frac{\mathbf{Cpu} + \mathbf{Mem} + \mathbf{Bw}}{\mathbf{3}}$$

$$\mathbf{IB_{host}} = \frac{\mathbf{1}}{\mathbf{3}}\left\{\frac{|\boldsymbol{Cpu} - \boldsymbol{Avg}|}{\mathbf{Avg}} + \frac{|\boldsymbol{Mem} - \boldsymbol{Avg}|}{\mathbf{Avg}} + \frac{|\boldsymbol{Bw} - \boldsymbol{Avg}|}{\mathbf{Avg}}\right\} \quad (\mathbf{5})$$

Cpu, Mem, and Bw are mention the percentage remaining resource capacity of a host; Avg is define as the average remaining capacity of the host and refers to imbalance impact of the host.

4. LOAD MEASUREMENT

Load distribution in-between the several physical machines and load balancing are very important tasks in cloud computing because multiple physical machines are used to serve the user's requests concurrently in a cloud environment. Due to inappropriate scheduling, circumstances may occur where some of the nodes are overloaded while remaining nodes do very little work or under loaded which decrease resource usages and system efficiency. So load in-between the several PM must be distributed equally or at any moment of time each node be supposed to perform around the same quantity of work. In following figure 3 shows datacenter with inappropriate allocation (unbalance datacenter) and with appropriate allocation (balance datacenter).

Load of PM is intended in terms of cup load, bandwidth or memory capacity. On the basis of these different load measurements methods are projected in literature. Wood et al. [11] projected a load calculation as follows-

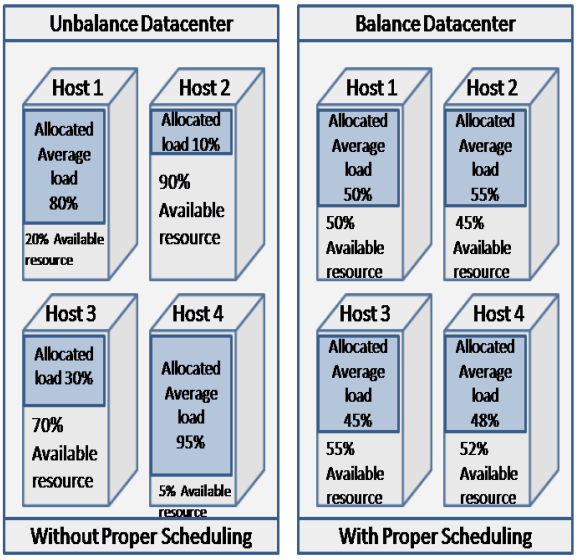

Fig.3 Load distribution in Datacenter

$$V = \frac{1}{(1-\text{CPUu})(1-\text{MEMu})(1-\text{BWu})}$$

Where $CPUu$, BWu and $MEMu$ are refers the average CPU, bandwidth and memory utilization of a physical machine.

Wenhong Tian et al. [6] to measure load of a physical machine in terms of average remaining capacity of Resources are calculated as given below.

$$\text{Host}_{\text{cap}} = \frac{K_1 * C_{\text{rem}} + K_2 * M_{\text{rem}} + K_3 * B_{\text{rem}}}{3} \qquad (6)$$

$$where \sum_i^3 K_i = 1 \qquad (7)$$

Here mention the average remaining capacity of the physical machine, are defined as the constant and Cpurem, Brem and Mrem are the available or remaining capacity of CPU, Bandwidth and Memory resources of a physical machine.

5. PROPOSED WORK

In this section Allocation Strategy is proposed for virtual machine allocation which efficiently solves the problems of server consolidation, minimizing the response time and balance resource utilization in the data center. When the cloud receives a VM request from user with resource requirements R, a new VM will be allocated on a PM in real time. There are no of policies to elect the PM to host the new VM. In this paper algorithms are introduced based on various policies. If all physical machines have the same available resource capacity, so the proposed algorithm is tradeoff approach.

5.1 Hosts Classification in Cloud Datacenter

In proposed algorithm the hosts are categorized according to their resource accessibility. If there are n types of resources, hosts can be divided into n! Indexes according to permutation. There are three types of resources Bandwidth, CPU, and Memory hosts are divided into 3! i.e. six indexes in which each array has particular types of hosts using the permutation and so on. Fig shows the six resources availability index based on three types of resources, i.e. Bandwidth, CPU, and Memory. Assume in first index has hosts which has resource availability in an order as B≥M≥C, second have B≥>=C≥M,

Algorithm 1: Datacenter creation Algorithm

Total number of Host = N

[1] If resources Type =3 (CPU, BW and MEM) then

[2] Total number of Host_Index = 6.

[3] Create the 6 empty Host_Index according to the resources type

[4] for i=1 to N Create the Host[i] and initialize them

[5] Add the Host[i] into the Host_Index as

[6] if Host[i] resources have CPU≥BW≥MEM then add it into the Host_Index which have resources capacity as CPU≥BW≥MEM.

[7] And so on.

[8] End if

[9] End for

[10] Sort the all Host_Index in descending order of available resources availability.

Third have M≥C≥B, fourth have M≥B≥C, fifth have C≥>=M≥B and sixth have C≥>=B≥M. Here C≥M≥B means, in these hosts the CPU availability is more than or equivalent to Memory availability and memory availability is more than or equal to bandwidth and this list is opposite to the list having resource capacity B≥M≥C and so on. Now hosts in these lists are sorted in descending (For proposed algorithm) order according to remaining resource capacities of hosts. The Algorithm-1 shows the host classification process in the cloud datacenter.

Host Classification In Cloud Datacenter Based On Available Resource						
Category	Host List					
B≥C≥M	H_1	H_2	H_3 .	.....	H_{n-1}	H_n
B≥M≥C	H_{n1+1}	H_{n1+2}	H_{n1+3}	.....	H_{n2-1}	H_{n2}
M≥C≥B	H_{n2+1}	H_{n2+2}	H_{n2+3}	.....	H_{n3-1}	H_{n3}
M≥B≥C	H_{n3+1}	H_{n3+2}	H_{n3+3}	.....	H_{n4-1}	H_{n4}
C≥B≥M	H_{n4+1}	H_{n4+2}	H_{n4+3}	.....	H_{n5-1}	H_{n5}
C≥M≥B	H_{n5+1}	H_{n5+2}	H_{n5+3}	.....	H_{n6-1}	H_{n6}
C=M=B	H_{n6+1}	H_{n6+2}	H_{n6+3}	.....	H_{n7-1}	H_{n7}

Fig. 4 Host Classification in Datacenter

6. FLOW DIAGRAM OF PROPOSED ALGORITHM

In FIG.4 the working flow diagram of the proposed algorithm. Let us assume that when a request (B≥C≥M) of virtual machine (already running or newly created)

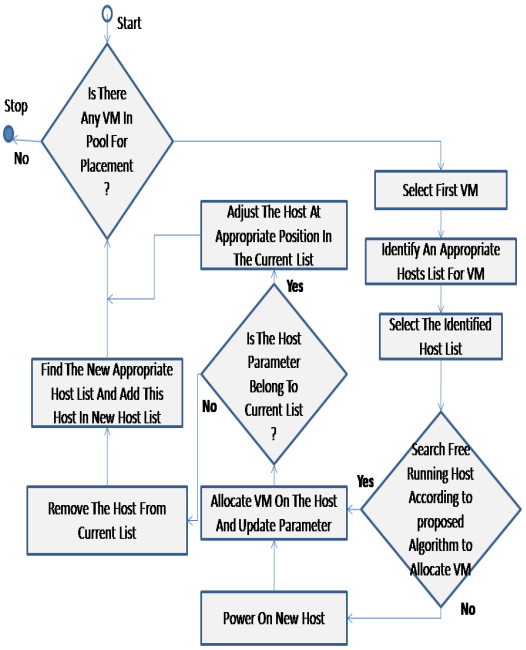

Fig .5. Flow graph of projected Method

For VM allocation arrives from the virtual machine pool, and then the virtual machine scheduler allocates this virtual machine on the fulfilling host in the index in which all hosts have available resource capacity in order as B≥C≥M. In other words, if a virtual machine having Bandwidth requirement greater than or equal to CPU requirement and the CPU must greater than or equal to Memory is placed on the fulfilling host in the index, in which all hosts have an available resource capacity of Bandwidth more than or equal to CPU and CPU more than or equal to Memory. The fulfilling host in the selected index is searched according to the proposed algorithm. If fulfilling, satisfying running host is not available to allocate the virtual machines, power ON the new host and allocate virtual machine (VM) on that host and update the host parameters. Host available capacity of each resource type does not belong to the current index, remove the host from the current index, find the new appropriate index, add the host at the appropriate place in the new index.

7. PROPOSED ALGORITHM

In case of proposed algorithm the hosts of each index in the datacenter are sorted in decreasing order of remaining capacity of resources. When ask for (new VM or already Running VM) for Virtual machine allocation, arrive at the data center, Virtual machine scheduler finds the appropriate list and place this Virtual machine on to the first satisfying host of the selected list.

System explanation-Fig.6 below shows the model for virtual machine task to host list present in the datacenter.

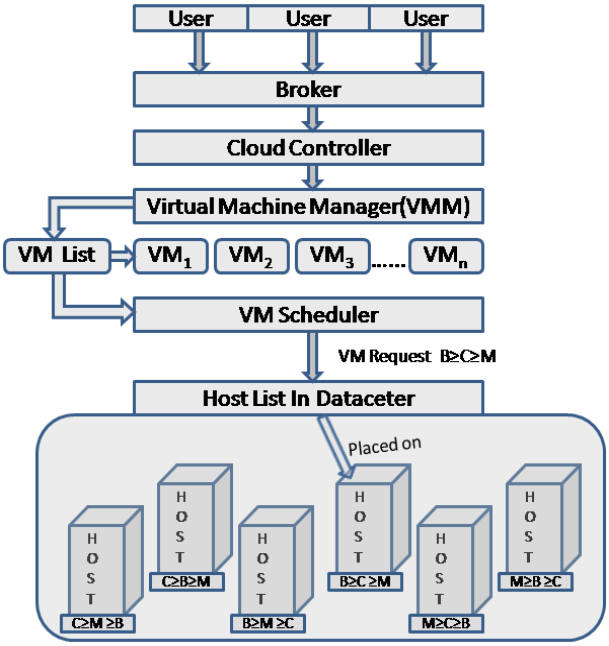

Fig. 6 Frame work of Proposed Algorithm

The Algorithm 2 shows a Virtual machine allocation using the proposed algorithm. If first host does not have required resource to satisfy in all dimensions, power ON the new PM, allocate the Virtual machine on the new physical machine. The update procedure is same for the algorithm which is given above in the flow graph working of the proposed model. According to the algorithm 2 the worst case time complexity of the proposed algorithm for m number of virtual machines will be □(□ □□□□), where n is the number of PMs, but according to □□□□3 response time (time required to search the destination host) complexity for a the VM will be constant i.e. □(1).

Algorithm 2: Proposed Algorithm

Input: Total number of Hosts n, Total number of Host_Index = 6,

VM_Index = 1 , Total number of VMs m.

[1] for each VM[i] in VM_Index

[2] Find an appropriate Host_Index as Selected_Host_Index.

[3] if first host of Selected_Host_Index satisfy the VM[i] requirement then

[4] Select the first host of Selected_Host_Index as a Selected_Host.

[5] else Power ON new Host as a Selected_Host.

[6] end if

[7] Allocate VM on the Selected_Host.

[8] Update the following parameters.

i. Selected_Host_CPU=Selected_Host_CPU-VM_CPU

ii. Selected_Host_MEM=Selected_Host_MEM-VM_MEM

iii. Selected_Host_BW= Selected_Host_BW-VM_BW

[9] if Selected_Host parameter does not belong to the current Selected_Host_List then

[10] eliminate the Selected_host from the Current Host_Index and find the new Host_Index to which Selected_Host belong as a Selected_Host_Index .

[11] Add the Selected_Host at appropriate position in the new Selected_Host_Index.

[12] else Adjust the Selected_Host in current Selected_Host_Index at appropriate position.

[13] end if

[14] end for

8. EXPERIMENTAL RESULTS

In this section, we provide simulation results for comparing three different scheduling algorithms Introduced in this paper with running 21 different size VMs on the 21 PMs of dissimilar capabilities. FIG.6 shows the mean imbalance impect of a Cloud datacenter. It can be seen that the proposed algorithm has lowest mean imbalance impect of a Cloud datacenter. The rise in imbalance impect means the rise in resource leak. In the case of proposed algorithms the physical machines are utilizing resources in a more balance manner then other algorithms

Experimental outcomes demonstrate that the projected algorithm superior then Balance Algorithm [7] and Greedy First Fit Algorithm [8] for all evaluated parameters in the case of physical machines of dissimilar capacities. In addition to this the proposed strategies provide good results in comparison to other approaches such as Modification Best Fit Decreasing Algorithm[10] and Efficient VM allocation Based on Binary search[9] for all evaluated parameters. The experimental results for two evaluated parameters (Imbalance level and Load distribution on physical machines) are shown below in graphs.

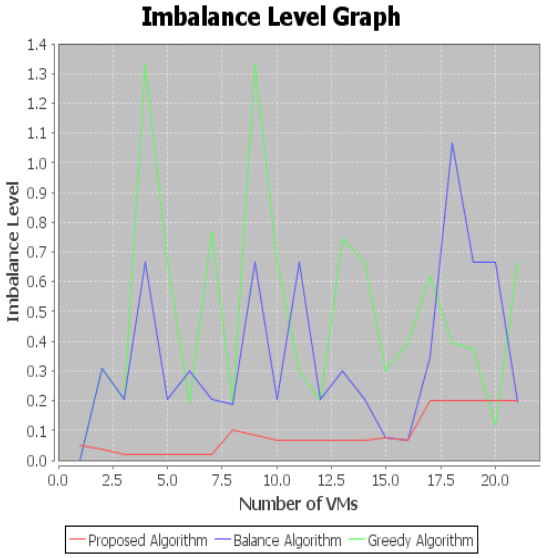

Fig. 7. Simulation Result for Imbalance Level

The figure 7 shows the load distribution (in terms of average remaining resource capacity of a physical machine) on physical machines of various algorithms. The load distribution illustrates the average available capacity of a physical machine, which is defined earlier. The graph presents that in the case of proposed algorithm the load on all physical machines are equally distributed, and in proposed algorithm, to allocate the 13 VMs it required less PMs in comparison to other algorithms Balance required 21 and Greedy required 20. So in case proposed algorithm unutilized PMs can be turned OFF, which be capable of given the server consolidation.

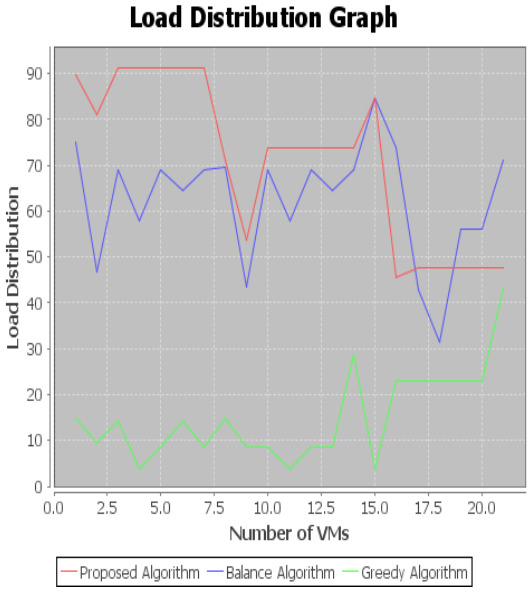

Fig.8.Simulation Result for Load Distribution on PMs.

9. CONCLUSION AND FUTURE WORK

This paper represents the applications of VM scheduling techniques to minimize the VM placement time or response time for user‘s request with measurable improvements in resource utilization by providing balance utilization of resources, server workload management and minimizing the required number of physical machines to for allocation of VMs. The proposed VM scheduling techniques present improved cost advantages to the clients as well as cloud vendors. The section for VM placement and load balancing is

also increasing the adaptability of the algorithms. Thus, it is concluded that proposed scheduling strategies can provide the efficient VM placement with minimum overhead time as well as efficient utilization of resources in the field of cloud computing.

The evaluation of presented proposed work is done using CloudSim simulation toolkit. So run time challenges can be resolved by applying the proposed algorithms in real world cloud environment. Another important factor for further investigation is use of upper and lower threshold values for utilization of resources. The use of higher threshold value for over loaded physical machines can provide the efficient load balancing and the use of lower threshold value for under loaded physical machines can provide the server consolidation.

10. REFERENCES

[1] Michael Miller, "Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online", 1st ed., USA: Que Publishing, 2008.

[2] R. Buyya, J. Broberg, A. Goscinski, "Cloud Computing: Principle and Paradigms", 1st ed., Hoboken: John Wiley & Sons, 2011.

[3] A. Weiss. "Computing in the Clouds", *netWorker*, 11(4): 16-25, ACM Press, New York, USA, Dec. 2007.

[4] Kalagiakos, P.; Karampelas, P., "Cloud Computing learning," in *5th International Conference on Application of Information and Communication Technologies (AICT)*, 2011, vol., no., pp.1-4.

[5] "Eucalyptus",[Online]available:http://www.eucalyptus.com/eucalyptuscloud

[6] W. Tian, Y. Zhao, Y. Zhong, M. Xu and C. Jing, "A dynamic and integrated load-balancing scheduling algorithm for Cloud datacenters", in Proc. *International Conference on Cloud Computing and Intelligence Systems (CCIS)*, Beijing : IEEE, 2011.

[7] X. Li, Z. Qian, R. Chi, B. Zhang, and S. Lu, "Balancing Resource Utilization for Continuous Virtual Machine Requests in Clouds", in Proc. *Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS)*, Palermo: IEEE, 2012.

[8] Subramanian S, Nitish Krishna G, Kiran Kumar M, Sreesh P4 and G R Karpagam, "An Adaptive Algorithm For Dynamic Priority Based Virtual Machine Scheduling In Cloud" *International Journal of Computer Science Issues (IJCSI)*, Vol. 9, Issue 6, No 2, November 2012.

[9] S. K. Mandal and P. M. Khilar, "Efficient Virtual Machine Placement for On-Demand Access to Infrastructure Resources in Cloud Computing", *International Journal of Computer Applications (IJCA)*,Vol. 68, No.12, April 2013.

[10] R. Buyya, A. Beloglazov, and J. Abawajy, "Energy-Efficient Management of Data Center Resources for Cloud Computing: A Vision, Architectural Elements, and Open Challenges", in proceedings *International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA)*, Las Vegas, USA, July 12-15, 2010.

[11] T.WOOD, P. Shenoy and A. Venkataramani, "Black-box and Gray-box Strategies for Virtual Machine Migration", in the proceedings *4th USENIX conference on Networked systems design & implementation(NSDI)*, Berkeley : ACM, 2007.

[12] Rodrigo N Calheiros, Rajiv Ranjan, Anton Beloglazov, Cesar AF De Rose, and Rajkumar Buyya, "Cloudsim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms", *Software: Practice and Experience*, 41(1):23{50, 2011.

[13] Bhathiya Wickremasinghe, Rodrigo N. Calheiros, and RajkumarBuyya, "CloudAnalyst: A CloudSim-based Visual Modeller for Analyzing Cloud Computing Environments and Applications", in: Proceedings *24th International Conference on Advanced Information Networking and Applications (AI NA)*, 2010.