Volume 3, Issue 12, December 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Review on Virtual Machine Management Techniques and Scheduling in Cloud Computing

Dilshad H. Khan, Prof. Deepak Kapgate, Prof. P.S Prasad
Department of C.S.E., GHRAET,
Nagpur University, Nagpur, India

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. Load Balancing 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.

II. VM SCHEDULING TECHNIQUES

Cloud computing technology virtualizes and offers many services across the network. It mainly aims at scalability, availability, throughput, and resource utilization. Emerging techniques focus on scalability and availability.

A. Migration Techniques:

To improve the utilization of cloud resources we use virtual machines. Virtual machine is a software implementation of a computing environment in which operating system or program can be installed and run. Live VM migration is the process of moving virtual machine from one physical host to other host without disturbing others. All migration techniques are trying to reduce total migration time and down time. Below we explain two migration techniques, which are used for reducing total migration time and down time [12].

1) *Pre-copy:* In pre-copy migration first transfer the memory contents to the target machine. After completing the Memory transfer processor states are transferred to destination.

2) *Post copy:* In post copy memory data are transferred after the processor states transfer. In order to reduce the total number of pages transfer during the migration, here we mention four different techniques,

- Adaptive memory compression,
- LRU and splay tree algorithm
- Live virtual machine migration using CPU scheduling
- Check point recovery trace and replay.

B. Resource Management Techniques:

Cloud computing referred to as the on demand technology because it offers dynamic and versatile resource allocation for reliable and warranted services in pay as-you-use manner to public. The resource allocation in cloud computing is nothing but integrating the cloud provider activities in order to utilize and allocate scarce resources. In cloud environments, effective resource allocation strategies are -

1) Linear Scheduling Strategy: In [20] Here scheduling algorithms mainly target on the distribution of the resources among the requestors which is able to maximize the chosen QoS parameters. The QoS parameter selected in this approach is the cost function. The scheduling algorithm is designed based on the tasks and the available virtual machines together and named LSTR scheduling strategy. This is often designed so as to maximize the resource utilization.

2) Match Making And Scheduling: In [21] it tells that the "Match making" is the first step and "scheduling" is second within the resource allocation in cloud environment. Matchmaking is that the method of allocating jobs associated with user requests to resources designated from the obtainable resource pool. Scheduling refers to determining the order in which jobs mapped to a selected resource are to be executed [21]. It additionally tells that there are some uncertainties that are associated with such type of "match making" and scheduling.

3) Just-In-Time Resource Allocation: To optimize resource usage and to reduce the number of idle resources, a perfect solution is to set a time interval and alter resources as persistently keeping with workload changes [22]. Within the limit of this interval resources are changed unceasingly in accordance with the modification in load, assuming we are able to continually overestimate the load. The limit of the interval is made too tiny. This extreme will make ensure that the optimum number of resources is always being used. Clearly, such as scheme is not possible since changing resources is not spontaneous. And it also makes some problems in the cost related aspects.

4) MiyakoDori(a mechanism for memory reuse): MiyakoDori [23], is a memory reusing mechanism to reduce the amount of transferred data in a live migrating system. When we are considering the case of dynamic VM consolidation, virtual machines may migrate back to the host where it was once executed and so the memory image in that host can be reused, thus contributing to shorter migration time and greater optimizations by VM placement algorithms.

C. Fault Tolerant Technique

While concerning on large scale system, fault tolerance is a very critical issue, since the cloud resources are extensively disseminated among diverse locations. This leads to a higher probability of failures while solving huge problems, thus the cloud service reliability could be relatively low. Therefore, providing an effective fault tolerance technique for a cloud system is mandatory.

D. Memory-aware cloud scheduling techniques

J. Ahn [15] proposes and evaluates two cluster-level virtual machine scheduling techniques for cache sharing and non-uniform memory accesses (NUMA) Affinity, which do not require any prior knowledge on the behaviors of VMs. For memory-aware scheduling, the cloud scheduler collects the cache behavior of each VM from computing nodes, and migrates VMs if such migration can potentially reduce the overall cache misses and the average memory access latencies by NUMA affinity in the cloud system [15]. In each computing node, a monitor checks "last level cache (LLC)" misses with hardware performance monitoring counters, and periodically sends the per-VM LLC miss and NUMA affinity information to the cloud scheduler. Based on the VM status information from all the nodes, the cloud scheduler makes global scheduling decisions.

E. Real-Time Scheduling

T. Cucinotta [16] proposes a mechanism for providing temporal isolation based on a CPU real time scheduling strategy. This allows not only to have control over the 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.

III. VM SCHEDULING ALGORITHM

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].

There are main two categories of scheduling algorithm. First is static scheduling algorithm and another is dynamic scheduling algorithm. Both have their own advantage and limitation. Dynamic scheduling algorithms have higher performance than static algorithm but have a lot of overhead compare to it. Static algorithms are mostly suitable for homogeneous and stable environments and can produce very good results in these environments. However, they are usually not flexible and cannot match the dynamic changes to the attributes during the execution time. Dynamic algorithms are more flexible and take into consideration different types of attributes in the system both prior to and during run-time. These algorithms can adapt to changes and provide better results in heterogeneous and dynamic environments. However, as the distribution attributes become more complex and dynamic. As a result some of these algorithms could become inefficient and cause more overhead than necessary resulting in an overall degradation of the services performance.

A) Gang scheduling Algorithm

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. Gang scheduling is a special case of scheduling parallel jobs in which tasks of jobs need to communicate very frequently. Gang scheduling involves high overhead since network status must be saved and then be restored when switching between tasks.

Moschakis et. al. [8] gives improved version of gang scheduling and performance and Cost evaluation of Gang Scheduling. Usually, the scheduling methods implemented in the scheduler aim for better response times and lower slowdowns, by minimizing unnecessary delays. The scheduler must also tend to the cost of the lease time of VMs aiming for a better cost-to-performance ratio [8]. The study takes into consideration both performance and cost while integrating mechanisms for job migration and handling of job starvation. The number of Virtual Machines (VMs) available at any moment is dynamic and scales according to the demands of the jobs being serviced. Results highlight that this scheduling strategy can be effectively deployed on Clouds. The number of VM's available at any moment is dynamically scales according to the demand of the job being served. For this purpose they applied "shortest Queue First (SQF) algorithm which dispatches the tasks to VMs with the shortest queue.

The research on gang scheduling has shown the potential of time sharing in improving throughput [17]. Migrating VM to a new data center is generally expensive, and also it does not eliminate starvation. Gang scheduling can be effectively applied in cloud computing environment both performance wise and cost-wise.

Response Time R_j, of a job j is the time interval between the arrival and the departure of the job. Its average is defined as [8]:

$$\text{Response Time } \sum_{J=1}^{n} R_j = (Rj / n)$$

- Where n is the total number of jobs.

B) Round Robin algorithm

In the round robin scheduling, Virtual Machines are dispatched to physical hardware in a FIFO manner but are given a limited amount of CPU time [6] called a time-slice or a quantum. If a process does not complete before its time quantum, the Virtual Machine execution is preempted and given to the next Virtual machine waiting in a queue. The preempted process is then placed at the back of the ready list. A time quantum is generally from 100-1000 milliseconds. So, the RR algorithm will allow the first VM in the queue to run until it expires its quantum (i.e. runs for as long as the time quantum), then run the next VM in the queue for the duration of the same time quantum. The RR algorithm is naturally pre-emptive. RR algorithm is one of the best scheduling algorithms that developed by many researchers

RR 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].

C) Genetic algorithm

Genetic algorithm is for scheduling sets of independent VM's, the objective of genetic algorithm is to minimize the make span. Initially in GA many individual solutions are (usually) randomly generated to form an initial population. The population size depends on the nature of the problem i.e. type and no of VM's to be run effectively on system. During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (VM's schedule likely to give effective response time) are typically more likely to be selected. The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover and mutation. This generational process is repeated until a termination condition has been reached i.e. a solution is found that satisfies minimum response time criteria .

GA will increase the cost of time, space, throughput and improve the quality of service of the entire .The goal of GA is to reduce the scheduled time of VM. Genetic algorithm provides both improved response time to VM via parallel execution. A state of the system and through genetic algorithm the migration cost becomes a problem.

D) 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.

Once we have chosen that host node, we calculate the difference between the new VM and the VMs residing at the host; then, we transfer only the difference to the destination host. Finally, at the destination host, we can reconstruct the new VM from the difference that was transferred and the contents of local VMs. Content based VM scheduling algorithm that can significantly reduced the network traffic associated with transfer of VM's from storage racks to host racks in cloud data center.

E) Adaptive Algorithm

K. Kumar [6] proposed adaptive algorithm which uses dynamic priority for nodes based on which the VM's are scheduled and assigned. Depending upon the priority values, the VM's to the nodes are scheduled, which varies dynamically based on their load factor. Priority of a node is assigned depending upon its capacity and the load factor. This algorithm strikes the right balance between performance and power efficiency as and when the virtual machines are assigned to the nodes, recalculation of their priorities takes place. The dynamic Priority concept leads to better utilization of the resources. Adaptive algorithm is an efficient algorithm for finding expected response time of each Virtual machine. It improve the throughput, achieves high bandwidth utilization and outage probability of the system.

F) Priority scheduling algorithm

The basic idea is straightforward; each Virtual Machine is assigned a priority, and priority is allowed to run. Equal-Priority instances are scheduled in FCFS order. Priorities are assigned based on the characteristics of VM's such as amount of workload, predicted execution time, user assigned priority. Internally defined priorities use some measurable quantities or qualities to compute priority of a VM. Priority once assigned to VM can be changed dynamically by using concept of aging i.e. here the priority of VM keeps increasing based on the total amount of time VM remains in ready queue waiting for execution. If priority of VM increased greater than the VM executing on physical hardware the executing VM preempts with VM having higher priority. Preemption of VM from physical hardware is also done when a VM is created or migrated to system having higher priority than VM executing on hardware.

Vignesh V et. al. [13] in their paper proposed improved priority scheduling algorithm using SJF policy. The shortest-Job-First (SJF) policy is used as a special case of general priority scheduling algorithm. An SJF algorithm is simply a priority algorithm where the priority is the inverse of the next CPU burst. That is, the longer the CPU burst, the lower the priority and vice versa [13].

It has High processor utilization, High throughput, Minimize turnaround time. It can be change its priority based on its age or execution history.

G) Efficient Resource Utilization Algorithm

R. Nivethitha [9] explains use of the massive pool of resources in terms of pay-as-you use policy. On demand the resources are delivers by the cloud through the use of network resources under different conditions. Based on their usage the effective utilization of resources the users will be charged. His named his proposed algorithm as "Effective Resource Utilization Algorithm (ERUA)" is based on 3-tier cloud architecture (Consumer, Service Provider and the Resource Provider) which benefits both the user (QoS) and the service provider (Cost) through effective schedule reallocation based on utilization ratio leading to better resource utilization.

Performance analysis made with the existing scheduling techniques shows that efficient resource utilization algorithm gives out a more optimized schedule and enhances the efficiency rate [9]. The service provider hires resources from the resource provider and creates Virtual Machine (VM) instances dynamically to serve consumers.

H) Deadline-Aware Algorithm

K. Parrott [10] explained a novel approach to optimize job deadlines when run in virtual machines by developing a deadline-aware algorithm that responds to job execution delays in real time and dynamically optimizes jobs to meet their deadline obligations. The algorithm intelligently schedules the jobs and learns over time about the missed deadlines under various conditions and tries to predict whether job would be meeting its deadline *di*, and if not then take appropriate measures to improve it chances in meeting deadline *di*.

This implies:

*virtual exec. time = (duration * overhead) + duration*

The main goal of the Deadline aware scheduler is to guarantee a start service time for a request. It does that by imposing a deadline on all I/O operations to prevent starvation of requests, reducing job deadline miss rate and increasing job throughput rate. Deadline-aware algorithm has flexible use and high utilization of the datacenter resources.

I) Renewable Energy Source provisioned algorithm

D. Hatzopoulos et. al. [11] in their paper explores the problem of virtual machine (VM) allocation in a network of cloud server facilities which are deployed in different geographical areas [11]. He addresses the problem of energy-efficient allocation in the system. The objective is to reduce the total cost of power consumption for the operator. Each request for a task to be executed in the cloud is associated with a VM request with certain resource requirements and a deadline by which it needs to be completed. The cloud provider has to create a VM with the resource requirements of the request and to execute the VM before the deadline. He proposes an online algorithm with given look-ahead horizon, in which the grid power price and pattern of output power of the RES are known a priori.

Renewable Energy Source provisioned algorithm applications leads to significant reduction of waste in bandwidth resource. This approach and design the metering method to observe the resource states. It determines its maximum request throughput.

IV. Comparison Of Different Algorithms

Serial number	Scheduling Algorithms	Environment	Throughput	Response time	Fault tolerant	Migration time	Resource Utilization	Performance
1	Gang Scheduling	Grid and Cloud Computing	√	√	√	√	√	√
2	Round Robin	Cloud Computing	√	√	X	X	√	√
3	Genetic Algorithm	Cloud computing	√	√	X	X	√	X
4	Content-Based VM Scheduling Algorithm	Cloud computing	X	X	√	X	√	X
5	Adaptive Algorithm	Grid and Cloud Computing	√	√	X	X	√	√
6	Priority scheduling algorithm	Cloud computing	√	√	X	√	√	√
7	Efficient Resource Utilization Algorithm	Cloud computing	√	√	√	X	√	√
8	Deadline-Aware Algorithm	Cloud computing	√	X	X	√	√	X
9	Renewable Energy Source provisioned	Cloud computing	√	X	X	X	√	X

V. CONCLUSION AND FUTURE WORK

In this paper, we surveyed multiple algorithms and techniques for virtual machine management for Cloud Computing. We discussed the challenges that must be addressed to provide the most suitable and efficient VM scheduling algorithms. We also discussed the advantages and disadvantages of these algorithms. Then, we compared the existing algorithms based on the performance matrices we discussed. Our research focuses on efficient use of Renewable Energy Source provisioned algorithm to VM management at different data centre's located at different geolocatable locations and also to consider the parameters like bandwidth available and latency delays before use of such scheduling algorithm. As our future work, we are planning to improve existing VM management algorithms to make it more suitable for multimedia services and application where long term connection between client and datacenter is applicable. Also to schedule Vm's such as to make Cloud environments more efficient in terms of storage utilization.

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.

