ISSN 2047-3338

# Minimizing Virtual Machine Migration for Efficient Resource Management in Green Clouds

Sukhvir Kaur[1] and Supriya Kinger[2]

[1,2]Computer Science and Engineering Department, Sri Guru Granth Sahib World University, Fatehgarh Sahib, Punjab, 140406, India

[1]kaur.sukh61@gmail.com, [2]supriya@sggswu.org

***Abstract*– Cloud computing refers to the Internet based development and utilization of computer technology, and hence, cloud computing can be described as a model of Internet-based computing. The primary aim of cloud computing is to provide efficient access to remote and geographically distributed resources. Load balancing is the major concern in the cloud computing environment. Cloud comprises of many hardware and software resources and managing these will play an important role in executing a client's request. In this present situation the load balancing algorithms built should be very efficient in allocating the request and also ensuring the usage of the resources in an intelligent way so that underutilization and overutilization of the resources will not occur in the cloud environment. The main goal of scheduling is to maximize the resource utilization i.e. energy reduction, power management and minimize processing time of the tasks using migration. In this research paper we present a generalized hybrid algorithm for efficient execution of tasks and comparison with earlier algorithm for optimization, FCFS and priority scheduling. We have specifically focused on improving computational cloud performance in terms of total processing time, power management and energy consumption. A simulation of proposed approach on Windows Azure platform has been conducted. Experimental results show that proposed algorithm performs efficiently. Results are compared with existing resource management algorithms in computational cloud environment.**

***Index Terms*– Cloud Computing, Load Balancing, Virtual Machine Migration, Scheduling, Green Clouds and Power Management**

## I. INTRODUCTION

CLOUD computing is a promising computing paradigm which is to support virtualization, scalable resource utilization, Para-virtualization and provide services such as Infrastructure as a Service, Software as a Service, Platform as a Service. Computer scientist predicting that cloud system is next generation operating system. Google engineers say a magnificent sentence maintains thousands of servers, warned that if power consumption continues to increase, power cost can easily overtake hardware cost by a large margin [1]. Consumer of cloud only needs internet connection. The burden of purchasing a new license copy, installing an application, update that copy monthly all burden can be removed, if users are cloud utilize. It can offer services on base of energy, power Pay-per use model. In office environments, computers, Monitors account for highest energy consumption after lighting. Power dissipation is also a major concern in portable battery operated devices that have rapidly increased [2]. Green computing is a new inclination, trend for high end computer.

To realize the full potential of cloud computing, cloud middleware needs to support various services such as security, uniform access, resource management, task Scheduling, application composition, and economic computation. Though, a range of essential services are to be integrated to accomplish a real cloud environment, among them scheduler is one of the most critical service component of the cloud middleware. Since, it is responsible for selecting best suitable virtual machines or computing resources with a goal of maximizing resource utilization and scheduling tasks, in a manner that meets user and task requirements, in terms of overall processing time, processing cost or any other constraints imposed upon by the user.

This paper focuses on various scheduling algorithms like FCFS, Generalized Priority Algorithm along with the concept of virtual migration. Virtualization, a technique to run several operating systems simultaneously on one physical server, has become a core concept in modern data centers, mainly driven by benefit of application isolation, resource sharing, fault tolerance, portability and cost efficiency.

Remaining paper is organized as follow: section second describes the Load balancing in Cloud Computing environment, section third describes the Virtualization and Virtual Machine, section fourth describes the literature survey, section fifth describes the literature survey, section sixth describes the results and simulation and finally section seventh describes the conclusion and future work.

## II. LOAD BALANCING IN CLOUD COMPUTING ENVIRONMENT

Load balancing in cloud computing provides an efficient solution to various issues residing in cloud computing environment set-up and usage. Load balancing must take into account two major tasks, one is the resource provisioning or resource allocation and other is task scheduling in distributed environment [3]. Efficient provisioning of resources and scheduling of resources as well as tasks will ensure:

a). Resources are easily available

b). Resources are efficiently utilized under condition of high/low load

c). Energy is saved in case of low load (i.e., when usage of cloud resources is below certain threshold)

d). Power consumption and Time consumptions get reduced

A typical Cloud modeled using Window Azure consists of following four entities datacenters, Hosts, Virtual Machines and Application as well as System Software. Datacenters entity has the responsibility of providing Infrastructure level Services to the cloud users. They act as a home to several Host Entities or several instances hosts' entities aggregate to form a single datacenter entity. Hosts in Cloud are physical servers that have pre-configured processing capabilities. Host is responsible for providing software level service to the Cloud users. Hosts have their own storage and memory. Processing capabilities of hosts is expressed in MIPS (million instructions per second). They act as a home to Virtual Machines or several instances of Virtual machine entity aggregate to form a Host entity. Virtual Machine allows development as well as deployment of custom application service models. They are mapped to a host that matches their critical characteristics like storage, processing, memory, software and availability requirements. Thus, similar instances of Virtual Machine are mapped to same instance of a host based upon its availability. Application and system software are executed on Virtual Machine on-demand. Class diagram of Cloud architecture illustrating relationship between the four basic entities is shown in Fig. 1.

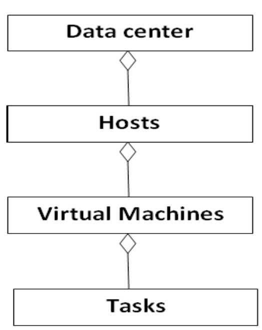


Fig. 1: Class Diagram of Cloud

## A. *Resource Allocation*

Resource provisioning is the task of mapping of the resources to different entities of cloud on demand basis. Resources must be allocated in such a manner that no node in the cloud is overloaded and all the available resources in the cloud do not undergo any kind of wastage (wastage of bandwidth or processing core or memory etc). Mapping of resources to cloud entities is done at two levels [4].

- *VM Mapping onto the Host*

Virtual machines reside on the host (physical servers). More than one instance of VM can be mapped onto a single host subject to its availability and capabilities. Host is responsible for assigning processing cores to VM. Provisioning policy define the basis of allocating processing cores to VM on demand. Allocation policy or algorithm must ensure that critical characteristics of Host and VM do not mismatch.

- *Application or Task Mapping onto VM*

Applications or tasks are actually executed on VM. Each application requires certain amount of processing power for their completion. VM must provide required processing power to the tasks mapped onto it. Tasks must be mapped onto appropriate VM based upon its configuration and availability.

- *Task Scheduling*

Task scheduling is done after the resources are allocated to all cloud entities. Scheduling defines the manner in which different entities are provisioned. Resource provisioning defines which resource will be available to meet user requirements whereas task scheduling defines the manner in which the allocated resource is available to the end user (i.e. whether the resource is fully available until task completion or is available on sharing basis). Task scheduling provides "Multiprogramming Capabilities" in cloud computing environment. Task scheduling can be done in two modes

a. Space shared

b. Time shared

Both hosts and VM can be provisioned to users either in space shared mode or time shared mode.

In space sharing mode resources are allocated until task does not undergo complete execution (i.e., resources are not preempted); whereas in time sharing mode resources are continuously preempted till task undergoes completion. Table 1 gives the comparison of resource allocation and task scheduling and specifies the issues resolved by each technique of load balancing. Based on resource provisioning and scheduling, four cases can be examined under different performance criteria so as to get efficient load balancing scheme [5].

*Case 1:* Hosts and VMs, both are provisioned in space sharing manner.

*Case 2:* Hosts and VMs, both are provisioned to VMs and tasks respectively in time sharing manner.

*Case 3:* Hosts are provisioned to VMs in space sharing manner and VMs are provisioned to tasks in time sharing manner.

*Case 4:* Hosts are provisioned to VMs in time sharing manner and VMs are provisioned to tasks in space sharing manner.

Table 1: Comparison of resource allocation and task

| Task | Sub Category | Issue Resolved | Provider Oriented | Customer Oriented |
|---|---|---|---|---|
| Resource Allocation | At Host Level<br>At VM Level | Efficient Utilization<br>Minimize Makespan<br>Ensure Availability | Yes | Yes |
| Task Scheduling | Space-Sharing<br>Time-Sharing | Minimize overall response time | No | Yes |

## III. VIRTUALIZATION AND VIRTUAL MACHINES

It is the technique that removes linking together the hardware and operating system. It directs to the source of the logical resources abstraction away from their physical resources to be more flexible, reduce costs and make a good improvement in business value. Essentially virtualization in cloud has so many different types, such as, network virtualization, server virtualization and storage virtualization. Server virtualization can be described as an associating of single physical resources to several logical partitions or representations. In a virtualized environment, computing environments can be produced in a forceful dynamic manner, enlarged, become smaller or go in a specified direction or manner as demand varies. Virtualization is therefore highly suitable to a dynamic cloud infrastructure, because it provides important advantages in isolation, manageability and sharing [6].

### *A) Types of Virtualization*

#### *1) Hardware virtualization*

Hardware virtualization or platform virtualization refers to the creation of a virtual machine that acts like a real computer with an operating system. Software executed on these virtual machines is separated from the underlying hardware resources. For example, a computer that is running Microsoft Windows may host a virtual machine that looks like a computer with the Ubuntu Linux operating system; Ubuntu-based software can be run on the virtual machine. In hardware virtualization, the host machine is the actual machine on which the virtualization takes place, and the guest machine is the virtual machine. The words host and guest are used to distinguish the software that runs on the physical machine from the software that runs on the virtual machine.

The software or firmware that creates a virtual machine on the host hardware is called a hypervisor or Virtual Machine Manager. Different types of hardware virtualization include:

*Full virtualization:* Almost complete simulation of the actual hardware to allow software, which typically consists of a guest operating system, to run unmodified.

*Partial virtualization:* Some but not the entire target environment is simulated. Some guest programs, therefore, may need modifications to run in this virtual environment.

*Para virtualization:* A hardware environment is not simulated; however, the guest programs are executed in their own isolated domains, as if they are running on a separate system. Guest programs need to be specifically modified to run in this environment.

Hardware-assisted virtualization is a way of improving the efficiency of hardware virtualization. It involves employing specially designed CPUs and hardware components that help improve the performance of a guest environment. Hardware virtualization is not the same as hardware emulation. In hardware emulation, a piece of hardware imitates another, while in hardware virtualization; a hypervisor (a piece of software) imitates a particular piece of computer hardware or the entire computer. Furthermore, a hypervisor is not the same as an emulator; both are computer programs that imitate hardware, but their domain of use in language differs.

#### *2) Desktop virtualization*

Desktop virtualization is the concept of separating the logical desktop from the physical machine. One form of desktop virtualization, virtual desktop infrastructure (VDI), can be thought as a more advanced form of hardware virtualization. Rather than interacting with a host computer directly via a keyboard, mouse, and monitor, the user interacts with the host computer using another desktop computer or a mobile device by means of a network connection, such as a LAN, Wireless LAN or even the Internet. In addition, the host computer in this scenario becomes a server computer capable of hosting multiple virtual machines at the same time for multiple users.

#### *3) Software virtualization*

- Operating system-level virtualization, hosting of multiple virtualized environments within a single OS instance.
- Application virtualization and workspace virtualization, the hosting of individual applications in an environment separated from the underlying OS. Application virtualization is closely associated with the concept of portable applications.
- Service virtualization, emulating the behavior of dependent (e.g., third-party, evolving, or not implemented) system components that are needed to exercise an application under test (AUT) for development or testing purposes. Rather than virtual zing entire components, it virtualizes only specific slices of

dependent behavior critical to the execution of development and testing tasks.

*4) Storage virtualization*

- Storage virtualization, the process of completely abstracting logical storage from physical storage
- Distributed file system, any file system that allows access to files from multiple hosts sharing via a computer network
- Virtual file system, an abstraction layer on top of a more concrete file system, allowing client applications to access different types of concrete file systems in a uniform way
- Storage hypervisor, the software that manages storage virtualization and combines physical storage resources into one or more flexible pools of logical storage [10]
- Virtual disk drive, a computer program that emulates a disk drive such as a hard disk drive or optical disk drive (see comparison of disc image software)

*5) Memory virtualization*

- Memory virtualization, aggregating random-access memory (RAM) resources from networked systems into a single memory pool
- Virtual memory, giving an application programs the impression that it has contiguous working memory, isolating it from the underlying physical memory implementation [7].

### *B) Virtual Machines*

A virtual machine (VM) is a software implementation of a machine (i.e. a computer) that executes programs like a physical machine. Virtual machines are separated into two major categories, based on their use and degree of correspondence to any real machine:

A *system virtual* machine provides a complete system platform which supports the execution of a complete operating system (OS). These usually emulate an existing architecture, and are built with either the purpose of providing a platform to run programs where the real hardware is not available for use (for example, executing software on otherwise obsolete platforms), or of having multiple instances of virtual machines lead to more efficient use of computing resources, both in terms of energy consumption and cost effectiveness (known as hardware virtualization, the key to a cloud computing environment), or both.

A *process virtual machine* (also, language virtual machine) is designed to run a single program, which means that it supports a single process. Such virtual machines are usually closely suited to one or more programming languages and built with the purpose of providing program portability and flexibility (amongst other things). An essential characteristic of a virtual machine is that the software running inside is limited to the resources and abstractions provided by the virtual machine—it cannot break out of its virtual environment [6].

## IV. LITERATURE SURVEY

Richa Sinha et al. [8] conclude that power consumption can be reduced by live migration of the virtual machine as required and by switching off idle machines. So, they proposed a dynamic threshold based approach for CPU utilization for host at data center. This consolidation works on dynamic and unpredictable workload avoiding unnecessary power consumption. They would not meet energy efficiency requirement but would also ensure quality of service to the user by minimizing the Service Level Agreement violation and also validate the proposed technique results with higher efficiency.

Prayook Jatesiktat et al. [9] in this paper, a new load balancing for IaaS cloud system called TPPC has been proposed. TPPC algorithm can reduce system power consumption by converge the number of booted physical machine to the level of system workload in dynamic workload environment. Inevitably, decreasing of power consumption comes fairly with a tradeoff that is the slight increasing of process time. In the future, there are many issues that can be explored. First, how to properly schedule the VM usage of the real CPU such that the better overlapped between VM executions is achieved. Thus, the respond time will be improved greatly. This can be done by the modification of VM aware knowledge into kernel scheduling. In addition, the better control of CPU affinity can be used to enhance the performance of cloud system services.

Jinhua Hu et al. [10] Says that this paper builds a model based on the concrete situations of cloud computing. It considers the historical data and current states of VM, uses tree structure to do the coding in genetic algorithm, proposes the correspondent strategies of selection, hybridization and variation also puts same control on the method so that it has better astringency. However in real cloud computing environment, there might be dynamic change in VMs, and there also might be an increase of computing cost of virtualization software and some unpredicted load wastage with the increase of VM number started on every physical machine. Therefore, a monitoring and analyzing mechanism is needed to better solve the problem of load balancing. This is also a further research subject.

Cloud is made up of massive resources. Management of these resources requires efficient planning and proper layout. While designing an algorithm for resource provisioning on cloud the developer must take into consideration different cloud scenarios and must be aware of the issues that are to be resolved by the proposed algorithm. Therefore, resource provisioning algorithm can be categorized into different classes based upon the environment, purpose and technique of proposed solution.

Amit Agarwal et al. [11] FCFS for parallel processing and is aiming at the resource with the smallest waiting queue time and is selected for the incoming task. Allocation of application-specific VMs to Hosts in a Cloud-based data center is the responsibility of the virtual machine provisioned component. The default policy implemented by the VM provisioned is a straightforward policy that allocates a VM to the Host in First-Come-First-Serve (FCFS) basis. The disadvantages of FCFS is that it is non preemptive. The shortest tasks which are at the back of the queue having to wait for the long task at the front to finish .Its turnaround and response is quite low. Customer defines the priority according to the user demand you have to define the parameter of cloudlet like size, memory, bandwidth scheduling policy etc. In the proposed strategy, the tasks are initially prioritized according to their size such that one having highest size has highest rank. The Virtual Machines are also ranked (prioritized) according to their MIPS value such that the one having highest MIPS has the highest rank. Thus, the key factor for prioritizing tasks is their size and for VM are their MIPS. This policy is performing better than FCFS and Round Robin scheduling [7].

## V. PROPOSED WORK

This section presents problem statement and a gives a brief discussion on proposed algorithm.

Virtualization technologies, on which Cloud Computing environment heavily rely on, provide the ability to transfer Virtual Machines between Physical nodes using live or offline migration. In cloud computing, the system should avoid wasting resources as a result of under-utilization and avoid lengthy response time as a result of over-utilization. Virtualized data centers make a lot of unnecessary migrations and cannot work in a large-scale cloud computing environment. The energy consumption and power consumption also increases as we increase the time for resource execution. So the time is the major focus for the utilization of resources as this increase the energy and power also increases.

This is first step that start system and configure all subsystem to process. Generate task module is a part of system that is used for generating tasks for execution on multiple systems network. Here user can create task and all are stored into their corresponding database. Generate systems module is a part of system that is used for generating system (node) for execution of generated task. Here user can create System and all are store into their corresponding database. Sorting module sorts all systems, so that high performance system can be allocated job earlier as compared to other systems. All tasks from queue are taken and a threshold based on RAM is generated. It is to be confirmed that the job queue has some job to execute. Distribute all tasks on the systems and calculate results from them. It execute until the system limit is not reached. Now if tasks are left than system creates a virtual machine that is the combination of free configuration of all systems in the network. Allocate task to VM, calculate result and publish results.

### A) Hybrid Approach for Resource Scheduling in Green Clouds

Task Scheduling problems in Cloud Computing are different from other scheduling problem for a particular task. Task Scheduling in Cloud Computing are different from another scheduling problem because computing services in Cloud Computing are offered through a SLA between cloud users and providers. The goal of cloud providers is to satisfy the user requirement and to increase its profit. In this, we

*Algorithm Used:*

1. START K=0;
2. Initialize Total Jobs=0;
3. Initialize Job Selection = True;
4. For Every Selected job Initialize Server Configuration = True
5. For k=1: Selected job
6. If System. Configuration. Meets. Job. Specification
7. Job .Allocated = True; k=k+1;
8. Apply Job. Sort (Set. Sorting. Priority = True)
9. If Overload. Occurred == True
10. Broadcast. System. Configuration = True
11. If Reply. Outcome == True
12. Selection Procedure = Reply. Address
13. Set VM=X Where x=Reply VM
14. If Job. Completed == True
15. Free VM
16. End

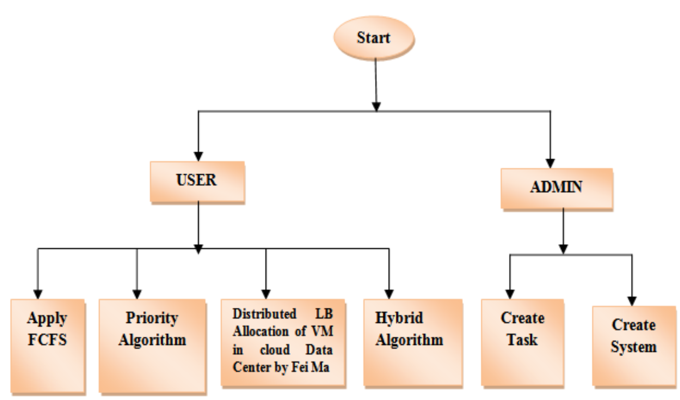


Fig. 2: System Model

introduced an administrator whose core function is to generated the users, different types of tasks and systems with different specification on which task performs. After the task is generated it is allocated to the system that matches with its requirement. Each task has some requirements like: Time, Energy Consumption, and Power Consumption.

We assumes centralized scheduling scheme; i.e., a master processor unit in collecting the entire task, will take charge of dispatching them to the system which is most suitable for that particular task. The systems generated by the administrator have its own specifications. These specify system features are RAM, Processor, Power consumption, Energy Consumption

## VI. RESULTS AND SIMULATION

The results simulation has taken place in Window Azure. To measure the performance we have included three parameters Time, Energy and Power.

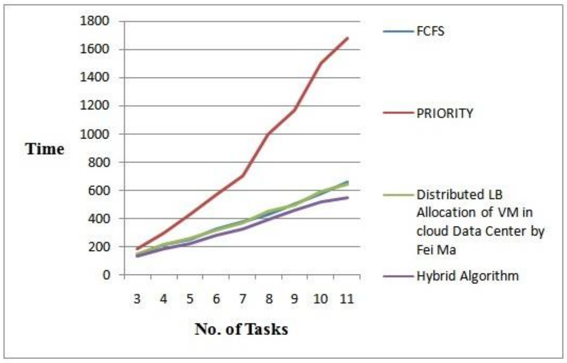


Fig. 2: Efficient Time Consumption

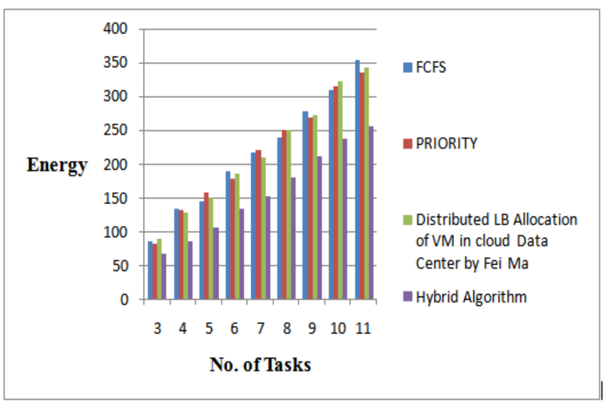


Fig. 3: Energy Consumption

From the above simulation, it has been concluded that Fig. 3 shows that proposed algorithm executes the maximum number of tasks in very less time i.e., 500 MS. Fig. 4 shows that Energy consumption of proposed algorithm is minimum i.e., 250 J from other methods. Fig. 5 shows that proposed algorithm has achieved a good power consumption rate than the previous algorithm and power consumption difference is about 400 KW. So it has been concluded that the proposed technique is can not only reduce Response time, conserve more energy but also achieves power management. This algorithm is more powerful in dynamic applications.

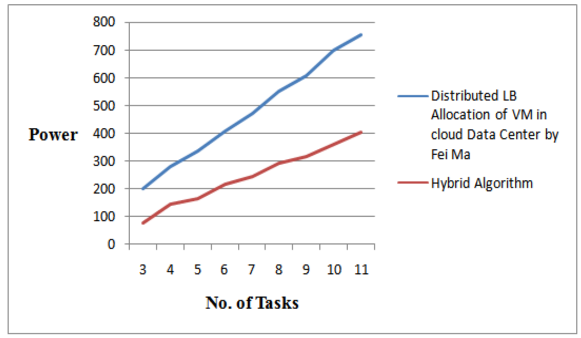


Fig. 4: Power Consumption by Proposed and Earlier Algorithm

## VII. CONCLUSION AND FUTURE WORK

From the experimental results illustrated above, it has been verified that proposed efficient algorithm is very efficient. Result shows that our system can achieve the better load balancing in a large scale cloud computing with less Virtual machine migration. Results show that it can not only reduce Response time, conserve more energy but also achieves power management. This algorithm is more powerful in dynamic applications. In the future one can consider different types of applications in cloud, and establish a more detailed load assess system. One can also measure utilization of resources to evaluate the effect of balancing in a cloud environment.

## REFERENCES


[1]. Glauco Estácio Gonçalves, et al." Balancing the Load across Virtual Links from Virtual Machine Requests in Distributed Clouds" 978-3-901882-50-0_c 2013 IFIP 506.

[2]. Vinay Kumar Kaushik, et al. "Load Balancing In Cloud Computing Using High Level Fragmentation of Dataset" International Conference on Cloud, Big Data and Trust 2013, Nov 13-15, RGPV.

[3]. Xinyu Zhang et al. "Load Balancing Algorithm based Virtual Machine Dynamic Migration Scheme for Datacenter Application with Optical Networks" 2012 7th International ICST Conference on Communications and Networking in China (CHINACOM) 978-1-4673-2699-5/12/$31.00 © 2012 IEEE.

[4]. Manan D. Shah et al. Reallocation and Allocation of Virtual Machines in Cloud Computing.

[5]. Jinhua Hu, et al. "Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment" 978-0-7695-4312-3/10 $26.00 © 2010 IEEE DOI 10.1109/PAAP.2010.65

[6]. Neeraj, et al. "Load Balance Based on Scheduling and Virtual Machine" © *2013, IJARCSSE* Volume 3, Issue 7, July 2013.

[7]. www.wikipedia.org

[8]. Richa Sinha et al. "Power Aware Live Migration for Data Center in Cloud using Dynamic Threshold" Vol2 (6), 2041-2046 IJCTA|NOV-DEC 2011

[9]. Prayook Jatesiktat et al. "Efficient Power Management on a Cloud System using Two Phases Power Convergence Algorithm 978-1-4673-1921-8/12/$31.00 © 2012 IEEE.

[10]. Jinhua Hu et al. "A scheduling on Load Balancing of Virtual Machine Resources in Cloud Computing Environment" 978-0-7695-4312-3/10 $26.00 © 2010 IEEE DOI 10.1109/PAAP.2010.65

[11]. Amit Agarwal, et al. "Efficient Optimal Algorithm of Task Scheduling in Cloud Computing Environment" International Journal of Computer Trends and Technology (IJCTT) – volume 9 number 7– Mar 2014

**Sukhvir kaur** is Student of M.Tech in the department of Computer Science at SGGSWU, Fatehgarh Sahib, Punjab. She has done B.Tech from BBSBEC, Fatehgarh Sahib Under Punjab Technical University, Jalandhar.

**Supriya** is currently working with SGGS World University, Fatehgarh, as Assistant Professor. She received her Master's in Technology degree in field of Computer Science from YMCA, India and Bachelor's in Technology degree in Computer Science from Kurukshetra University, India. Her areas of interest include Parallel and Distributed Processing, Cloud Computing and Software Engineering. Currently she is doing research in field of „Energy Efficient Clouds". She has more than 8 years of experience in UG & PG teaching and research. She has attended and organized a number of workshops and conferences and has presented papers in field of Green Clouds, Component Based Software Engineering, Information Security and Networks. She is a life member of ISTE.