ISSN No. 0976-5697

Volume 8, No. 5, May – June 2017

International Journal of Advanced Research in Computer Science

SURVEY REPORT

Available Online at www.ijarcs.info

A Survey on various Load Balancing Algorithms in Cloud Computing

Divyashree Nath
Dept. of Computer Science Engg
University Institute of Technology, RGPV
Bhopal, (M.P) India

Uday Chourasia
Dept. of Computer Science Engg
University Institute of Technology, RGPV
Bhopal, (M.P) India

Shikha Agarwal
Dept. of Computer Science Engg
University Institute of Technology, RGPV
Bhopal, (M.P) India

Abstract: In modern era cloud computing has bloomed into the greatest platform which provides storage of data in very lower cost and available for all time over the internet. But the cloud computing also has more critical issues such as security, load balancing and fault tolerance ability. In this paper the focus is on various Load balancing approaches. The Load balancing can be defined as the process of smart distribution of load to the different nodes of a server so as to attain optimal resource utilization. The process of Load balancing is required when a single node is overloaded, in such a situation, the process of load balancing the load is evenly distributed to the other ideal nodes. The load balancing algorithms have broadly been classified into Static load balancing and Dynamic load balancing. In this paper a survey of various load balancing algorithms are presented.

Keywords: cloud computing, load balancing, virtualization, load balancing algorithm, virtual machine.

I. INTRODUCTION

Cloud Computing [1], in the present scenario, can be treated both as a platform as well as a type of application. Cloud computing as a platform helps in the configuration and reconfiguration of servers, where the servers are the physical machines or virtual machines. Also via cloud computing, through the internet various applications are modified and given access to the clients, these applications are hosted in data centers which make use of large servers.

The cloud computing has become the easiest way to exploit the Internet and is a facade for the colossal amounts of infrastructure it conceals. But Cloud computing vastly varies from traditional computing models because it is more scalable and the services are provided through various levels according to the client's need, in which the scale and usage of the resources is dynamically configurable [2].

Cloud computing is a technique in distributed computing that concentrates on catering service to a broad range of subscribers with different needs by providing software and hardware infrastructure in a virtualized means through the internet [3]. It incorporates virtualization, distributed computing, networking, web software services. The concept of cloud computing has largely been concentrated on the interest of users in exploiting distributed, parallel and virtualization of the computing systems in the present scenario. It has been seen as a feasible solution for providing easy and cheap access to outsourced IT resources and facilities. With the help of virtualization, cloud computing will be capable to approach a wider client base with same amount of physical infrastructure but with different computational requirements.

A. Types of Cloud Deployments

A cloud deployment model represents a specific type of cloud environment, primarily distinguished by ownership, size, and access. There are four common cloud deployment models:

1. *Public cloud*- This type of cloud is generally used by the public or for a huge industry which may possibly be owned by an organization which is selling cloud facilities. Here, customer has no visibility or control of access, to the hosting site of computing infrastructure. The computing infrastructure can be shared between many organizations**.**
2. *Private cloud*- In this type of deployment computing infrastructure is run exclusively for the utilization of an organization. The cloud is possibly maintained by the same organization or any other third party. This has more security features in comparison to public clouds with added cost. Some clouds are externally hosted as well but a specialized third party hosts the cloud infrastructure. Therefore, Private clouds can be off-premises or on-premises. Externally hosted private clouds are cheaper in comparison to on-premise private clouds.
3. *Hybrid cloud*- Hybrid clouds are those in which different types of clouds are brought together as single unit with their own specialized deployment qualities intact. The best example is Cloud Bursting. In Cloud bursting, for trivial needs the organization utilizes their own computing infrastructure, but for larger load demands they access the cloud. This helps to manage a sudden increase in computing processes without much hindrance. Hybrid cloud may provides application portability and proprietary or standardized access to applications and data.
4. *Community cloud*- Community cloud is where the main function of is to support a common function or aim. Here the

cloud infrastructure has been shared by many organizations that support a specific community with a common goal.

B. Various Services Provided by Cloud

1. IAAS -Infrastructure as a service: Infrastructure as a service provides hardware related facilities which makes use of the canons of cloud computing. This offers many facilities such as virtual-machine, virtual storage; virtual infrastructure, raw block storage, load balancer, file storage, IP addresses, firewalls, disk image library, virtual LANs and the suppliers provide the required resources as the demand arise, from their data centers which consists of large bundles. The IaaS service provider is also responsible for the maintenance of the overall infrastructure. [4]

2. PAAS -Platform as a Service: In this a computing platform is provided by the cloud supplier, which also includes the execution environment for programming language, DB, OS and servers. The application developers can apply operations on their software as required through a cloud platform without the need to look after the cost of purchasing and maintaining the basic software and hardware requirements. The service provider has the sole responsibility for maintaining the cloud infrastructure and as well as the enabling of the OS.

3. SAAS -Software as a service : In SAAS the cloud providers gives the complete software package to the clients. Clients can access the software application hosted by the providers and the payment is done on the basis of pay as you use. Moreover in SAAS, clients are offered with access to DB and application softwares. SAAS manages the user interface as well as the applications with the help of complete operating. The providers of cloud are responsible for the maintenance of the infrastructure and platforms related to the applications. Since SAAS has a pay as you use policy it is called "on-demand software". SAAS suppliers will generally charge the clients a subscription fee for the initial infrastructure cost.

C. Cloud computing architecture

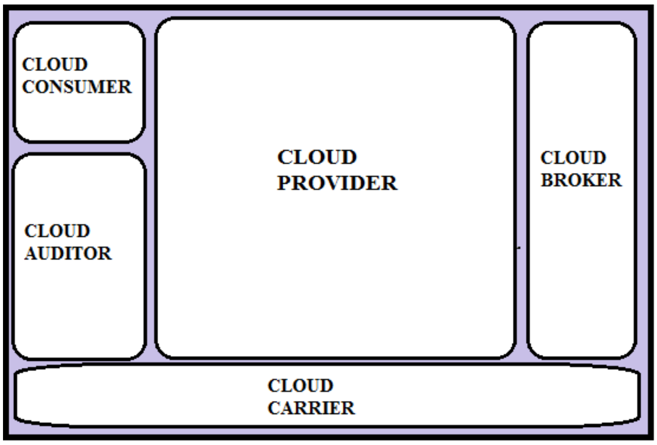

FIG 1: Cloud Computing Architectures

A cloud computing reference architecture [5] defines five major actors as shown in the figure 1, namely: cloud consumer, cloud provider, cloud auditor, cloud broker and cloud carrier where each actor is an entity which can be a person or an entity that participate in the process. These actors can be defined as follows:

1. *Cloud consumer*-A person or organization that maintains a business relationship with, and uses service from, Cloud Providers.

2. *Cloud provider*-A person, organization, or entity responsible for making a service available to interested parties.

3. *Cloud auditor*-A party that can conduct independent assessment of cloud services, information system operations, performance and security of the cloud implementation.

4. *Cloud broker*- It helps in the management of the use, performance and delivery of cloud services, and negotiates terms between Cloud Providers and Cloud Consumers.

5. *Cloud carrier*-An intermediary that provides connectivity and transport of cloud services from Cloud Providers to Cloud Consumers.

II. LOAD BALANCING

With the ever growing demand for cloud computing, the amount of data that is being processed in cloud computing has become colossal. Since the requests of the clients can be randomly to any nodes, therefore the load on each node may also vary i.e. some nodes are overloaded whereas some nodes are under loaded which may directly affect the efficiency of cloud services. Therefore, some kind of mechanism for smartly sharing the load is needed to ensure that every computing resource is distributed for the best utilization of the resources. Load Balancing can be defined as a method for distribution of workload across multiple nodes or a cluster of nodes, which may help to achieve optimal resource utilization, minimum response time, maximum throughput, and avoid overload [6]. It is a mechanism through which the dynamic local workload is evenly distributed across all the nodes in the entire cloud datacenter to achieve a high user satisfaction and the best possible resource utilization, thereby improving the overall performance. Load balancing also ensures that for every client process the computing resources are distributed efficiently and in a fair manner which further prevents bottlenecks and fail-over.

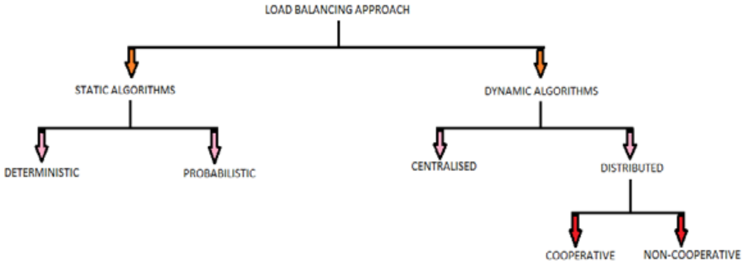

FIG 2: Taxonomy Of load Balancing Algorithm

Load balancing algorithms can be broadly classified[7] [8] into two types: Static algorithms and Dynamic algorithms. In Static Scheduling the allocation of tasks to the processors is done before the program execution begins i.e. during compile time[9]. Scheduling decision is generally based on various information such as task execution time and processing resources, which are assumed to be known during, compile time [9]. These methods are non-preemptive in nature. The goal of a static scheduling method is to reduce the overall execution time as soon as possible. Static algorithms are further classified into Deterministic and heuristic on the basis of prior knowledge they have about the nature of the processes [10]. Dynamic load balancing, on the other hand, is based on the redistribution of processes among the processors during the run time. This redistribution is achieved by the transferring of tasks to the lightly loaded ones from the heavily loaded processors. It is especially useful when the requirements of processes are not known before hand and the primary goal of these algorithms is the maximization of the resource utilization. The most important drawback of the dynamic load balancing algorithms is that, there are a major run-time overheads caused by the transfer of information regarding load among the processors and also tedious decision-making processes for the selection processors for job transfers and the communication delays associated with the task relocation. The dynamic load balancing algorithms can be classified into, centralized and distributed depending on whether the responsibility for the task resides on a single processor (centralized) or the work involved in making decisions should be shared among different processors [11]. The most important feature of making decisions centrally is the lower number of overheads because of its simplicity. Whereas the centralized algorithms suffer from the problem of bottleneck and single point failure because if the central node fails the whole system crashes. Distributed load balancing algorithms are better equipped for such problems. Again distributed dynamic scheduling can be classified into: cooperative or non-cooperative. The latter is simple where individual processors act alone as autonomous entities and make independent scheduling decisions without regard to the effect of their decision on the rest of the system. Whereas in cooperative scheduling algorithms, each processor has the awareness to carry out its own task to achieve a common unified goal for the system [12], [13].

III. LITERATURE SURVEY

The algorithms can be clearly understood by looking at the broad classifications of the algorithms, which are as follows:

A. Static Load Balancing Algorithm

These algorithms are used where there is low variation in load and the given algorithm gives no heed to the current state of the node. The various types of static load balancing are described below:

1. *Round Robin algorithm* [16][17] Round Robin is one of the those algorithms, for which preceding states of the nodes are not taken into account. Robin Method for job allocation is used, therefore it is very simple. The starting node is selected at random and the job is given to all the other nodes evenly in Round Robin Method. The main benefit of this algorithm is that it does not need any interprocess communication, therefore no need for overheads

2. *Weighted Round Robin Algorithm* [18] Here each node assigned has a specific weight. Based on the nodes weight, they would get the requests. If all nodes are equal, then the node is indicated to traffic.

3. *Random biased Algorithm* [19] It randomly assigns the selected jobs to the nodes. The algorithm is very simple but it does not take into consideration whether the nodes are overloaded or under loaded. Hence, this may result in the selection of a node which is heavily loaded and the job may have to wait for a long time before being served.

4. *Min-Min Algorithm* [20] The smallest task is assigned to the fastest resource. The task is removed from the set and same process is repeated. The method is simple. Does not consider the existing load on a resource.

5. *Max Min Algorithm* [21] This is very similar to the Min-Min algorithm. Here, those jobs having more execution time are executed first. This reduces the makespan. But smaller jobs have to wait for longer for bigger jobs to be completed.

6. *Weighted Least Connection* [22] Assigns tasks to the node having least number of connections. Balances load efficiently. Processing speed and storage capacity are not considered.

B. Dynamic Load Balancing Algorithm

This type of algorithm is based on the redistribution of processes among the processors during the run time. This redistribution is achieved by the transferring of tasks to the lightly loaded ones from the heavily loaded processors. It is especially useful when the requirements of processes are not known before hand and the primary goal of these algorithms is the maximization of the resource utilization.

1. *Central Manager Algorithm* In this algorithm a central processor will be choosing a slave processor for assigning of a job. This chosen slave processor will generally be the processor which will be minimally loaded. This information is send to remote processors and in this way all the processors keep updating their load information to the central manager. The central processor is able to gather all slave processor’s load information, and takes the allotment decisions based on the system load information taking the best possible decision.

2. *Ant Colony Optimization Based Load Balancing Algorithm* [24]This is inspired from the behavior of the ants where they follow the degree of pheromones and gather to the place which has most amount of food. The output is that, other ants are more likely to follow the path which is the shortest to a rich food source. The positive feedback eventually leads to all the ants following a single path. This idea is used to mimic this behavior with "simulated ants" walking around to get an optimal solution.

Table: 1 Summary of Load Balancing Techniques.

Sr. No.	Algorithm	Nature	Description	Advantages	Disadvantages
1.	**Round Robin[16][17]**	Static	The requests are allocated to the VMs(Virtual Machine) in circular manner with the first request randomly allocated.	The workload is equally distributed	The processing time for each request is not considered
2.	**Weighted Round Robin[18]**	Static	The VMs are ordered on the basis of their processing capacity	Resource Utilization is better.	The processing time for each request is not considered

3.	**Min-Min Algorithm [19]**	Static	The fastest resource is given the smallest task.	Performs better with load of small execution time.	The method may lead to starvation.
4.	**Throttled Load Balancing Algorithm [20]**	Dynamic	The state of each VM (busy/idle) is recorded & maintained. Request is processed only when match is found or else queue.	The load is shared evenly among the VMs	The load on current VM is not considered during allocation
5.	**Central Load Balancer [21]**	Static	VM allocation is based on priority which is calculated using CPU speed and memory capacity of VM.	Can work in heterogeneous environment.	Priority is fixed and there can be bottleneck problem.
6.	**Biased Random Sampling [22]**	Dynamic	The sampling walk starts at a specific node of a virtual graph and moves to a randomly chosen neighbour. The last node in the walk is selected for the load allocation.	Decentralized method so suitable for large network systems.	Not suitable for dynamic environment.
7.	**Opportunisti-c Load Balancing Algorithm [23]**	Static	The unexecuted tasks are sent to available nodes in a random order without regard to the nodes current situation	The overheads are less in number	Since no heed is given to the execution time, the task will be processed in a slower manner thus resulting in bottlenecks despite some of the nodes being free
8.	**Ant Colony Optimization [24]**	Dynamic	Based on actions of ants and seeking an optimal path in collecting their food.	Distributes the workload among nodes in efficient and optimal job scheduling is achieved.	Time consuming and large number of overheads.
9.	**Max-Min Load Balancing Algorithm [25]**	Static	Jobs with larger execution time are executed first.	Improves efficiency by increasing concurrent execution.	Execution that takes maximum time need to wait for long time.
10.	**User-Priority based Min-Min algorithm [26]**	Static	All the processes are divided into two groups based on priority and the group with higher one is executed first using min-min algorithm.	This also considers the priority of users and also the makespan	Not much heed is given to the deadline of each task
11.	**Honey bee Foraging [27]**	Dynamic	Decentralized nature inspired load balancing technique	Works well with heterogeneous resources	Increase in resources does not guarantee the throughput.
12.	**Weighted Least Connection [28]**	Dynamic	Assigns tasks to the node having least number of connections.	Balances load efficiently.	Processing speed and storage capacity are not considered.
13.	**Fractal-based Load Balancing Algorithm [29]**	Dynamic	Dynamic load balancing algorithm based on VM migration. Timing of VM migration is determined through load forecasting method.	Unnecessary migrations are not triggered in peak load.	Based on a threshold value.
14.	**Active Clustering balancing Algorithm [30]**	Dynamic	Grouping nodes together.	Similar nodes are grouped together.	The performance is poor when there is an increase in variety of nodes.

IV. BALANCING OF LOAD THROUGH VM MIGRATION

A. Virtualization

With virtualization [14], applications and infrastructure are independent, allowing servers to be easily shared by many applications where applications are running virtually anywhere in the world. This is possible as long as the application is virtualized. Vitalizing an application for the cloud means packaging the bits of the application with everything it needs to run, including pieces such as a database, a middleware and an operating system. This self-contained unit of virtualized application can then run anywhere in the world. With an increasing interest in SOA, which focus predominantly on ways of developing, publishing, and integrating application logic and/or resources as services, and the constantly growing solutions for virtualization have lead to the concept of cloud. Cloud computing represents a new type and specialized distributed computing paradigm, providing better use of distributed resources, while offering dynamic, flexible infrastructures and QoS guarantees. The load balancing problem can be divided into two sub problems[15]:

1. Submission of new task for VM provisioning and placement of VMs on host.

2. Reallocation/migration of VMs.

B. VM Migrations

A VM migration mechanism is a method in which with the help of virtualization the load at each server is adjusted. Whenever the server is under loaded or overloaded the host migrates the VM to another server where it continues with its execution. This process of transferring VMs is called VM migration which may occur numerous times during task execution for efficient Resource utilization. Migrations do not hinder the working of applications. During the process VM is migrated along with its current state of execution so that when it reaches the destination it can resume its execution from where it was left. After the completion of the process the required result is provided to the user through the data center where the user had submitted his task. Migration helps in the load balancing of the entire system.

In hardware virtualization, set of VMs are carved out from the physical hardware pieces. In a virtualization environment, a central hypervisor or any other central authority allocates resources like CPU and memory to VMs. Unlike the name suggests virtual machines do not compose of a physical interface or have a physical box or shell or anything to encapsulate and move around together.

In VM migration, these VMs are being moved by the system administrators between physical servers or other hardware pieces. In order to facilitate this, a new kind of migration has been evolved which is named as Live VM Migrations. This involves moving these virtual machines without hindering the VM processing or shutting the client system. Modern services often provide live migration feature which makes it much easier for moving virtual machines without much effort into other administrative work.

Cloud provider does not wish to compromise with the QoS requirements of its users but simultaneously it wants to reduce the operation cost by efficiently utilizing the available resources and turning off the underutilized server.

To achieve all these migration of VM is necessary. So we have to decide:

- When to migrate
- Which VM to migrate and
- Where to migrate

The migration is generally based on energy or on the basis of load. If any server gets overloaded on the basis of load or energy, then the VMs are migrated.

VM migrations can help in the reduction of energy consumption by:

(α) Bringing down the number of active servers by consolidation of tasks to lesser number of PMs with the help of virtualization.

(β) Shifting the processes from heavily-loaded servers to lesser loaded servers would drastically reduce the energy consumption of individual servers to better efficiency.

V. CONCLUSION

In this paper, we have surveyed many different types of algorithms that help in efficient load balancing of the servers. This helps to provide greater levels of fault tolerance and higher scalability.

Load balancing mainly has two functions: first, it directs the data traffic to multiple nodes respectively that reduces the response time for the users extensively; second, it transfers the load from heavily loaded nodes to those that are comparatively free since this improves the resource utilization of the entire system as well as each of the nodes. The strategy of allocation differs for different application environments e.g. the sites of e-commerce needs the CPU idle because these calculations need large processing while, on the other hand, the database applications that need to write and read frequently will need I/O idle nodes.

We have listed the various pros and cons of many load balancing algorithm. The addressing of challenges to these algorithms is important because this will help in invention of more efficient load balancing algorithms in future.

VM migration acts as an essential component of load balancing because VMs needs to be shifted to deal with the problems of either over utilization or under utilization of servers. The above mentioned algorithm not only provides balance but also helps in efficient utilization of resources, better throughput and quicker response time.

VI. REFERENCES

[1] Peter Mell, Timothy Grance. The NIST Definition of Cloud Computing (Draft). NIST. 2011.

[2] Nuaimi, K. Mohamed, N., Nuaimi, M, AI-Jaroodi, A survey of load balancing in cloud computing: challenges and algorithms, IEEE conference proceedings Network cloud computing and applications (NCCA), London, Dec 2012, pp 137-142.

[3] Kc gounda, Anurag Patro, Dines Dwivedi and Nagaraj Bhat ”Virtualization approaches in cloud computing” International Journal of Computer Trends and Technology (IJCTT), Vol. 12, Issue 4, June 2014.

[4] Rajesh Bose, Murari Krishna Saha and Debabrata Sarddar,” Fog Computing Made Easy with the Help of Citrix and Billboard Manager”, International Journal of Computer Applications, Volume 121 – No.7, July 2015.

[5] B. Wickremasinghe, CloudAnalyst: A CloudSim-based tool for modelling and analysis of large scale cloud computing environments, MEDC project report, 22(6), 2009, 433-659.

[6] P.Mathur, "Cloud Computing: new challenge to the entire computer industry", 1stInternational conference on parallel, distributed and grid computing, 2010, pp978-1- 4244-767.

[7] U.Chatterjee, "A Study on Efficient Load Balancing Algorithms in Cloud computing Environment", International Journal of Current Engineering and Technology, Vol.3, 11 November 2013

[8] Nuaimi, K. Mohamed, N., Nuaimi, M, AI-Jaroodi, A survey of load balancing in cloud computing: challenges and algorithms, IEEE conference proceedings Network cloud computing and applications (NCCA), London, Dec 2012, pp 137-142.

[9] X. Evers , "A Literature Study on Scheduling in Distributed Systems ",1992.

[10] ZenonChaczko, VenkateshMadadevan, ShahrzadAslanzadehand, ChristopherMedermind (2011),"Availability and Load Balancing in Cloud Computing", Vol.14.pp.138-140.

[11] Thomas L. Casavant, Jon G. Kuhl, "A Taxonomy of Scheduling in General-Purpose Distributed Computing Systems ", IEEE Trans, on

[12] X. Evers, W. H. CSG, CR. B. SG, I. S. Herschberg, D. H. J. Epema, and J. F. C. M. de Jongh, A literature study on scheduling in distributed systems, Delft University of Technology, 1992.

[13] Thomas L. Casavant, and Jon G. Kuhl, A Taxonomy of Scheduling in General-Purpose Distributed Computing Systems, IEEE Trans, on Software Eng., 14(2), 1988, 141-154.

[14] Lazaros Gkatzikis, Iordanis Koutsopoulos, "Migrate or Not? Exploiting Dynamic Task Migration in Mobile Cloud Computing Systems", IEEE Wireless Communications 2013, pp. 24-32.

[15] Raghavendra Achar, P. Santhi Thilagam, Nihal Soans, P. V. Vikyath, Sathvik Rao and Vijeth A. M. , 2013 Annual IEEE India Conference (INDICON), "Load Balancing in Cloud Based on Live Migration of Virtual Machines "

[16] A.khiyait, H.Bakkali, M.Zbakh, D.Kettani, Load Balancing Cloud Computing: state of art", University Mohammed V Souissi Rabat Morocco, 2012.

[17] Jasmin James, Dr. Bhupendra Verma, "Efficient VM Load Balancing Algorithm for a Cloud Computing Environment ", International Journal on Computer Science and Engineering (IJCSE) , Vol. 4 No. 09 Sep 2012 , pp. 1658-1663.

[18] Qi Zhang, Lu Cheng, Raouf Boutaba; Cloud computing: sate-of-art and research challenges; Published online: 20th April 2010, Copyright : The Brazillian Computer Society 2010.

[19] Isam Azawi Mohialdeen , "Comparative Study of Scheduling Algorithms in Cloud Computing Environment ", Journal of Computer Science , 2013, pp. 252-263.

[20] H.Mahalle, P.Kaveri, V.Chavan, "Load Balancing on Cloud Data Centers", international Journal of Advance Research in computer Science and Software Engineering, vol. 3, Jan. 2013.

[21] Dharmesh Kashyap, Jaydeep Viradiya(Nov 2014),"A Survey of Various Load Balancing Algorithms in Cloud Computing",Vol.3,Issue.11,pp.115-119.

[22] Lee, R. and B. Jeng, "Load-balancing tactics in cloud," International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), October 2011, IEEE, pp. 447-454.

[23] Isam Azawi Mohialdeen , "Comparative Study of Scheduling

Algorithms in Cloud Computing Environment ", Journal of Computer Science , 2013, pp. 252-263.

[24] Clinton Dsouza, Gail-Joon Ahn and Marthony Taguinod," Policy-Driven Security Management for Fog Computing: Preliminary Framework and A Case Study", IEEE IRI 2014, No. 13, pp 16-23, August 2014.

[25] M.Aruna , D. Bhanu, R.Punithagowri , "A Survey on Load Balancing Algorithms in Cloud Environment ", International Journal of Computer Applications, Volume 82 – No 16, November 2013 , pp. 39-43.

[26] Huankai Chen, Professor Frank Wang, Dr Na Helian, Gbola Akanmu, "User-Priority Guided Min-Min Scheduling Algorithm For Load Balancing in Cloud Computing", IEEE, 2013.

[27] Dharmesh Kashyap, Jaydeep Viradiya(Nov 2014),"A Survey of Various Load Balancing Algorithms in Cloud Computing",Vol.3,Issue.11,pp.115-119.

[28] Lee, R. and B. Jeng, "Load-balancing tactics in cloud," International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC), October 2011, IEEE, pp. 447-454.

[29] Haozheng Ren, Yihua Lan, Chao Yin, "The Load Balancing Algorithm in Cloud Computing Environment", 2nd International

Conference on Computer Science and Network Technology, IEEE, 2012, pp. 925-928.

[30] Martin Randles, David Lamb, A. Taleb-Bendiab , "A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing ", IEEE 24th International Conference on Advanced Information Networking and Applications Workshops , 2010, pp. 551-556.