IJCSN International Journal of Computer Science and Network, Volume 3, Issue 1, February 2014
ISSN (Online) : 2277-5420
www.IJCSN.org


# Load Balancing Techniques : Major Challenges in Cloud Computing - A Systematic Review

**[1]Jasobanta Laha, [2]Rabinarayan Satpathy , [3]Kaustuva Dev**

**[1,2,3] Computer Science., Biju Patnaik University of Technology (BPUT)**
**Rourkela, Odisha, 769001, India**

**Abstract -** Cloud Computing is an emerging area in the field of information technology (IT). Load balancing is one of the main challenges in cloud computing. It is a technique which is required to distribute the dynamic workload across multiple nodes to ensure that no single node is overloaded. Load balancing techniques help in optimal utilization of resources and hence in enhancing the performance of the system. The goal of load balancing is to minimize the resource consumption which will further reduce energy consumption and carbon emission rate that is the dire need of cloud computing. This determines the need of new metrics, energy consumption and carbon emission for energy-efficient load balancing in cloud computing. This paper mainly focused on the concept of load balancing technique in cloud computing, the existing load balancing techniques and also discusses the different qualitative metrics or parameters like performance, scalability, associated overhead etc.

***Keywords -*** *Load Balancing, Green Computing, Carbon Emission, Dynamic Load Balancing, Workload and Client aware policy (WCAP), ACCLB, CARTON, VectorDot.*

## 1. Introduction

Load balancing is one of the major issues in cloud computing [1]. It is a mechanism which distributes the dynamic local workload evenly across all the nodes in the whole cloud. This will avoid the situation where some nodes are heavily loaded while others are idle or doing little work. It helps to achieve a high user satisfaction and resource utilization ratio. Hence, this will improve the overall performance and resource utility of the system. It also ensures that every computing resource is distributed efficiently and fairly [2]. It further prevents bottlenecks of the system which may occur due to load imbalance. When one or more components of any service fail, load balancing helps in continuation of the service by implementing fair-over, i.e. in provisioning and de-provisioning of instances of applications without fail. It also ensures that every computing resource is distributed efficiently and fairly [1] [3]. Consumption of resources and conservation of energy are not always a prime focus of discussion in cloud computing. However, resource consumption can be kept to a minimum with proper load balancing which not only helps in reducing costs but making enterprises greener [4] [2]. Scalability which is one of the very important features of cloud computing is also enabled by load balancing. Hence, improving resource utility and the performance of a distributed system in such a way will reduce the energy consumption and carbon footprints to achieve Green computing [4] [5] [6].

## 2. Load Balancing

Load balancing is the process of improving the performance of the system by shifting of workload among the processors. Workload of a machine means the total processing time it requires to execute all the tasks assigned to the machine [7]. Load balancing is done so that every virtual machine in the cloud system does the same amount of work throughout therefore increasing the throughput and minimizing the response time [8]. Load balancing is one of the important factors to heighten the working performance of the cloud service provider. Balancing the load of virtual machines uniformly means that anyone of the available machine is not idle or partially loaded while others are heavily loaded. One of the crucial issue of cloud computing is to divide the workload dynamically. The benefits of distributing the workload includes increased resource utilization ratio which further leads to enhancing the overall performance thereby achieving maximum client satisfaction [9].

### 2.1 Why Balancing in Cloud Computing

Load balancing in clouds is a mechanism that distributes the excess dynamic local workload evenly across all the nodes. It is used to achieve a high user satisfaction and resource utilization ratio [10], making sure that no single node is overwhelmed, hence improving the overall performance of the system. Proper load balancing can help in utilizing the available resources optimally, thereby minimizing the resource consumption. It also helps in implementing fail-over, enabling scalability, avoiding bottlenecks and over-provisioning, reducing response time etc.

Load balancing is also needed for achieving *Green computing* in clouds [11]. The factors responsible for it are:

### 2.1.1 Limited Energy Consumption

Load balancing can reduce the amount of energy consumption by avoiding over hearting of nodes or virtual machines due to excessive workload [11].

### 2.1.2 Reducing Carbon Emission

Energy consumption and carbon emission are the two sides of the same coin. Both are directly proportional to each other. Load balancing helps in reducing energy consumption which will automatically reduce carbon emission and thus achieve Green Computing [11].

## 3. Load Balancing: It's Goal

The goals of load balancing are:

- To improve the performance substantially
- To have a backup plan in case the system fails even partially
- To maintain the system stability
- To accommodate future modification in the system

## 4. Categories of Load Balancing Algorithms

Depending on who initiated the process, load balancing algorithms can be of three categories:

### 4.1 Sender Initiated

If the load balancing algorithm is initialized by the sender

### 4.2 Receiver Initiated

If the load balancing algorithm is initiated by the receiver

### 4.3 Symmetric

It is the combination of both sender initiated and receiver initiated.
Depending on the current state of the system, load balancing algorithms can be divided into two categories:

**Static Algorithm**

Static algorithms divide the traffic equivalently between servers. By this approach the traffic on the servers will be disdained easily and consequently it will make the situation more imperfectly. This algorithm, which divides the traffic equally, is announced as round robin algorithm. However, there were lots of problems appeared in this algorithm. Therefore, weighted round robin was defined to improve the critical challenges associated with round robin. In this algorithm each servers have been assigned a weight and according to the highest weight they received more connections. In the situation that all the weights are equal, servers will receive balanced traffic [12].

**Dynamic Algorithm**

Dynamic algorithms designated proper weights on servers and by searching in whole network a lightest server preferred to balance the traffic. However, selecting an appropriate server needed real time Communication with the networks, which will lead to extra traffic added on system. In comparison between these two algorithms, although round robin algorithms based on simple rule, more loads conceived on servers and thus imbalanced traffic discovered as a result [12]. However; dynamic algorithm predicated on query that can be made frequently on servers, but sometimes prevailed traffic will prevent these queries to be answered, and correspondingly more added overhead can be distinguished on network.

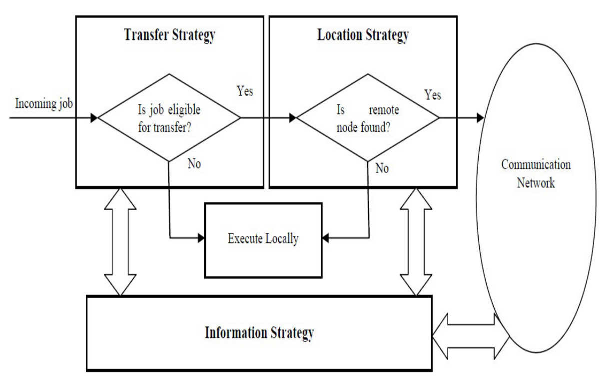


**Fig. 1** Interaction among components of a Dynamic Load Balancing Algorithm

## 5. Policies or Strategies in Dynamic Load Balancing

The different policies in dynamic load balancing are:

### 5.1 Transfer Policy

The part of the dynamic load balancing algorithm which selects a job for transferring from a local node to a remote node is referred to as Transfer policy or Transfer strategy.

### 5.2 Selection Policy

It specifies the processors involved in the load exchange (processor matching)

### 5.3 Location Policy

The part of the load balancing algorithm which selects a destination node for a transferred task is referred to as location policy or Location strategy.

### 5.4 Information Policy

The part of the dynamic load balancing algorithm responsible for collecting information about the nodes in the system is referred to as Information policy or Information strategy.

### 5.5 Load estimation Policy

The policy which is used for deciding the method for approximating the total work load of a processor or machine is termed as Load estimation policy.

### 5.6 Process Transfer Policy

The policy which is used for deciding the execution of a task that is it is to be done locally or remotely is termed as Process Transfer policy.

### 5.7 Priority Assignment Policy

The policy that is used to assign priority for execution of both local and remote processes and tasks is termed as Priority Assignment Policy.

### 5.8 Migration Limiting Policy

The policy that is used to set a limit on the maximum number of times a task can migrate from one machine to another machine.

## 6. Existing Load Balancing Techniques in Cloud Computing

### 6.1 Decentralized Content Aware Load Balancing

H.Mehta et al. [13] proposed a new content aware load balancing policy named as workload and client aware policy (WCAP). It uses a unique and special property (USP) to specify the unique and special property of the requests as well as computing nodes. USP helps the scheduler to decide the best suitable node for the processing the requests.

This strategy is implemented in a decentralized manner with low overhead. By using the content information to narrow down the search, this technique improves the searching performance and hence overall performance of the system. It also helps in reducing the idle time of the computing nodes hence improving their utilization.

### 6.2 Server-Based Load Balancing for Internet Distributed Services

A. M. Nakai et al. [14] proposed a new server based load balancing policy for web servers which are distributed all over the world. It helps in reducing the service response times by using a protocol that limits the redirection of requests to the closest remote servers without overloading them. A middleware is described to implement this protocol. It also uses a heuristic to help web servers to endure overloads.

### 6.3 Join-Idle-Queue

Y. Lua et al. [15] proposed a Join- Idle-Queue load balancing algorithm for dynamically scalable web services. This algorithm provides largescale load balancing with distributed dispatchers by, first load balancing idle processors across dispatchers for the availability of idle processors at each dispatcher and then, assigning jobs to processors to reduce average queue length at each processor. By removing the load balancing work from the critical path of request processing, it effectively reduces the system load, incurs no communication overhead at job arrivals and does not increase actual response time.

### 6.4 A Lock-Free Multiprocessing Solution for LB

X. Liu et al. [16] proposed a lock-free multiprocessing load balancing solution that avoids the use of shared memory in contrast to other multiprocessing load balancing solutions which use shared memory and lock to maintain a user session. It is achieved by modifying Linux kernel. This solution helps in improving the overall performance of load balancer in a multi-core environment by running multiple load-balancing processes in one load balancer.

### 6.5 Scheduling Strategy on Load Balancing of Virtual Machine Resources

J. Hu et al. [17] proposed a scheduling strategy on load balancing of VM resources that uses historical data and current state of the system. This strategy achieves the best load balancing and reduced dynamic migration by using a genetic algorithm. It helps in resolving the issue of load-imbalance and high cost of migration thus achieving better resource utilization.

### 6.6 Central Load Balancing Policy for Virtual Machines

A.Bhadani et al. [18] proposed a Central Load Balancing Policy for Virtual Machines (CLBVM) that balances the load evenly in a distributed virtual machine/cloud computing environment. This policy improves the overall performance of the system but does not consider the systems that are fault-tolerant.

### 6.7 LBVS: Load Balancing Strategy for Virtual Storage

H. Liu et al. [19] proposed a load balancing virtual storage strategy (LBVS) that provides a large scale net data storage model and Storage as a Service model based on Cloud Storage. Storage virtualization is achieved using an architecture that is three-layered and load balancing is achieved using two load balancing modules. It helps in improving the efficiency of concurrent access by using replica balancing further reducing the response time and enhancing the capacity of disaster recovery. This strategy also helps in improving the use rate of storage resource, flexibility and robustness of the system.

### 6.8 A Task Scheduling Algorithm Based on Load Balancing

Y. Fang et al. [20] discussed a two-level task scheduling mechanism based on load balancing to meet dynamic requirements of users and obtain high resource utilization. It achieves load balancing by first mapping tasks to virtual machines and then virtual machines to host resources thereby improving the task response time, resource utilization and overall performance of the cloud computing environment.

### 6.9 Honeybee Foraging Behavior

This algorithm is[21] derived from the behavior of honey bees for finding and reaping food. There is a class of bees called the forager bees which forage for food sources, upon finding one, they come back to the beehive to advertise this using a dance called waggle dance. The display of this dance, gives the idea of the quality or quantity of food and also its distance from the beehive. Scout bees then follow the foragers to the location of food and then began to reap it. They then return to the beehive and do a waggle dance, which gives an idea of how much food is left and hence results in more exploitation or abandonment of the food source.

In case of load balancing, as the webservers demand increases or decreases, the services are assigned dynamically to regulate the changing demands of the user. The servers are grouped under virtual servers (VS), each VS having its own virtual service queues. Each server processing a request from its queue calculates a profit or reward, which is analogous to the quality that the bees show in their waggle dance. One measure of this reward can be the amount of time that the CPU spends on the processing of a request. The dance floor in case of honey bees is analogous to an advert board here. This board is also used to advertise the profit of the entire colony. Each of the servers takes the role of either a forager or a scout.

### 6. 10 Biased Random Sampling

M. Randles et al. [22] investigated a distributed and scalable load balancing approach that uses random sampling of the system domain to achieve self-organization thus balancing the load across all nodes of the system. Here a virtual graph is constructed, with the connectivity of each node (a server is treated as a node) representing the load on the server. Each server is symbolized as a node in the graph, with each indegree directed to the free resources of the server.

Regarding job execution and completion,

- Whenever a node does or executes a job, it deletes an incoming edge, which indicates reduction in the availability of free resource.

- After completion of a job, the node creates an incoming edge, which indicates an increase in the availability of free resource.

The addition and deletion of processes is done by the process of random sampling. The walk starts at any one node and at every step a neighbor is chosen randomly. The last node is selected for allocation for load. Alternatively, another method can be used for selection of a node for load allocation, that being selecting a node based on certain criteria like computing efficiency, etc. Yet another method can be selecting that node for load allocation which is under loaded i.e. having highest in degree. If b is the walk length, then, as b increases, the efficiency of load allocation increases.

We define a threshold value of b, which is generally equal to log n experimentally. A node upon receiving a job, will execute it only if its current walk length is equal to or greater than the threshold value. Else, the walk length of the job under consideration is incremented and another neighbor node is selected randomly.

When, a job is executed by a node then in the graph, an incoming edge of that node is deleted. After completion of the job, an edge is created from the node initiating the load allocation process to the node which was executing the job. Finally what we get is a directed graph. The load balancing scheme used here is fully decentralized, thus making it apt for large network systems like that in a cloud.

### 6.11 Active Clustering

Active Clustering works on the principle of grouping similar nodes together and working on these groups. The performance of the system is enhanced with high resources thereby in-creasing the throughput by using these resources effectively. It is degraded with an increase in system diversity [23]. - A node initiates the process and selects another node called the matchmaker node from its neighbors satisfying the criteria that it should be of a different type than the former one.

- The so called matchmaker node then forms a connection between a neighbor of it which is of the same type as the initial node.
- The matchmaker node then detaches the connection between itself and the initial node.

The above set of processes is followed iteratively.

### 6.12 ACCLB (Load Balancing Mechanism Based on ant Colony and Complex Network Theory)

Z. Zhang et al. [24] proposed a load balancing mechanism based on ant colony and complex network theory in an open cloud computing federation. It uses small-world and scale-free characteristics of a complex network to achieve better load balancing. This technique overcomes heterogeneity, is adaptive to dynamic environments, is excellent in fault tolerance and has good scalability hence helps in improving the performance of the system.

### 6.13 Two-phase load balancing algorithm (OLB + LBMM)

S.-C. Wang et al. [25] proposed a two- phase scheduling algorithm that combines OLB (Opportunistic Load Balancing) and LBMM (Load Balance Min-Min) scheduling algorithms to utilize better executing efficiency and maintain the load balancing of the system. OLB scheduling algorithm, keeps every node in working state to achieve the goal of load balance and LBMM scheduling algorithm is utilized to minimize the execution time of each task on the node thereby minimizing the overall completion time. This combined approach hence helps in an efficient utilization of resources and enhances the work efficiency.

### 6.14 Event-Driven

V. Nae et al. [26] presented an event driven load balancing algorithm for real-time Massively Multiplayer Online Games (MMOG). This algorithm after receiving capacity events as input, analyzes its components in context of the resources and the global state of the game session, thereby generating the game session load balancing actions. It is capable of scaling up and down a game session on multiple resources according to the variable user load but has occasional QoS breaches.

### 6.15 CARTON

R. Stanojevic et al. [27] proposed a mechanism for cloud control named as CARTON that unifies the use of LB and DRL. LB (Load Balancing) is used to equally distribute the jobs to different servers so that the associated costs can be minimized and DRL (Distributed Rate Limiting) is used to make sure that the resources are distributed in a way to keep a fair resource allocation. DRL also adapts to server capacities for the dynamic workloads so that performance levels at all servers are equal. With very low computation and communication overhead, this algorithm is simple and easy to implement.

### 6.16 Compare and Balance

This algorithm [28] uses the concept of compare and balance to reach an equilibrium condition and manage unbalanced system's load. On the basis of probability (no. of virtual machine running on the current host and whole cloud system).The current node selects randomly a node and compares the load with itself.

### 6.17 VectorDot

A. Singh et al. [29] proposed a novel load balancing algorithm called VectorDot. It handles the hierarchical complexity of the data-center and multidimensionality of resource loads across servers, network switches, and storage in an agile data center that has integrated server and storage virtualization technologies. VectorDot uses dot product to distinguish nodes based on the item requirements and helps in removing overloads on servers, switches and storage nodes.

## 7. Qualitative Metrics for Load Balancing

In cloud computing, load balancing is required to distribute the dynamic local workload evenly across all the nodes. It helps to achieve a high user satisfaction and resource utilization ratio by ensuring an efficient and fair allocation of every computing resource. Proper load balancing aids in minimizing resource consumption, implementing fail-over, enabling scalability, avoiding bottlenecks and over-provisioning etc[30][31]. The different qualitative metrics or parameters that are considered important for load balancing in cloud computing are discussed as follows:

**i) Throughput:** The total number of tasks that have completed execution is called throughput. A high throughput is required for better performance of the system.

**ii) Associated Overhead:** The amount of overhead that is produced by the execution of the load balancing algorithm. Minimum overhead is expected for successful implementation of the algorithm.

**iii) Fault tolerant:** It is the ability of the algorithm to perform correctly and uniformly even in conditions of failure at any arbitrary node in the system.

**iv) Migration time:** The time taken in migration or transfer of a task from one machine to any other machine in the system. This time should be minimum for improving the performance of the system.

**v) Response time:** It is the minimum time that a distributed system executing a specific load balancing algorithm takes to respond.

**vi) Resource Utilization:** It is the degree to which the resources of the system are utilized. A good load balancing algorithm provides maximum resource utilization.

**vii) Scalability:** It determines the ability of the system to accomplish load balancing algorithm with a restricted number of processors or machines.

**viii) Performance:** It represents the effectiveness of the system after performing load balancing. If all the above parameters are satisfied optimally then it will highly improve the performance of the system

## 8. Load Balancing Challenges In The Cloud Computing

Although cloud computing has been widely adopted. Research in cloud computing is still in its early stages, and some scientific challenges remain unsolved by the scientific community, particularly load balancing challenges [32].

### 8.1 Automated Service Provisioning

A key feature of cloud computing is elasticity, resources can be allocated or released automatically. How then can we use or release the resources of the cloud, by keeping the same performance as traditional systems and using optimal resources?

### 8.2 Virtual Machines Migration

With virtualization, an entire machine can be seen as a file or set of files, to unload a physical machine heavily loaded, it is possible to move a virtual machine between physical machines. The main objective is to distribute the load in a datacenter or set of datacenters. How then can we dynamically distribute the load when moving the virtual machine to avoid bottlenecks in Cloud computing systems?

### 8.3 Energy Management

The benefits that advocate the adoption of the cloud is the economy of scale. Energy saving is a key point that allows a global economy where a set of global resources will be supported by reduced providers rather that each one has its own resources. How then can we use a part of datacenter while keeping acceptable performance?

### 8.4 Stored Data Management

In the last decade data stored across the network has an exponential increase even for companies by outsourcing their data storage or for individuals, the management of data storage or for individuals, the management of data storage becomes a major challenge for cloud computing. How can we distribute the data to the cloud for optimum storage of data while maintaining fast access?

### 8.5 Emergence of Small Data Centers for Cloud Computing

Small datacenters can be more beneficial, cheaper and less energy consumer than large datacenter. Small providers can deliver cloud computing services leading to geo-diversity computing. Load balancing will become a problem on a global scale to ensure an adequate response time with an optimal distribution of resources.

## 9. Conclusion

Load balancing is one of the major challenges in cloud computing. It is a mechanism which distributes the dynamic local workload evenly across all the nodes in the whole cloud. This will avoid the situation where some nodes are heavily loaded while others are idle or doing little work. It helps to achieve a high user satisfaction and resource utilization ratio. Hence, this will improve the overall performance and resource utility of the system. It also ensures that every computing resource is distributed efficiently and fairly. With proper load balancing, resource consumption can be kept to a minimum which will further reduce energy consumption and carbon emission rate which is a dire need of cloud computing. Existing load balancing techniques that have been discussed mainly focus on reducing associated overhead, service response time and improving performance etc. but none of the techniques has considered the energy consumption and carbon emission factors. But still there are many existing issues like Load Balancing, Virtual Machine Migration, Server Consolidation, Energy Management, etc. which have not been fully addressed. Central to these issues is the issue of load balancing, that is required to distribute the excess dynamic local workload evenly to all the nodes in the whole Cloud to achieve a high user satisfaction and resource utilization ratio.

## References

[1] B. P. Rima, E. Choi, and I. Lumb, “A Taxonomy and Survey of Cloud Computing Systems”, Proceedings of 5th IEEE International Joint Conference on INC, IMS and IDC, Seoul, Korea, August 2009, pages 44-51.

[2] A. M. Alakeel, “A Guide to dynamic Load balancing in Distributed Computer Systems”, International Journal of Computer Science and Network Security (IJCSNS), Vol. 10, No. 6, June 2010, pages 153-160.

[3] B. P. Rimal, E. Choi, and I. Lumb, “A Taxonomy, Survey, and Issues of Cloud Computing Ecosystems, Cloud Computing: Principles, Systems and Applications”, Computer Communications and Networks, Chapter 2 , pages 21-46, DOI 10.1007/978-1-84996-241-42, Springer – V erlagLondonLimited, 2010.

[4] R. Mata-Toledo, and P. Gupta, “Green data center: how green can we perform”, Journal of Technology Research, Academic and Business Research Institute, Vol. 2, No. 1, May 2010, pages 1-8.

[5] S. Kabiraj, V. Topka, and R. C. Walke, “Going Green: A Holistic Approach to Transform Business”, International Journal of Managing Information Technology (IJMIT), Vol. 2, No. 3, August 2010, pages 22-31.

[6] J. Baliga, R. W. A. Ayre, K. Hinton, and R. S. Tucker, “Green Cloud Computing: Balancing Energy in Processing, Storage, and Transport”, Proceedings of the IEEE, Vol. 99, No. 1, January 2011, pages 149-167.

[7] Sandeep Sharma, Sarabjeet Singh, Meenaksshi Sharma, “Performance Analysis of Load Balancing Algorithms”, World Academy of Science, Engineering and Technology, 2008.

[8] Hisao Kameda, EL-Zoghdy Said Fathyy and Inhwan Ryuz Jie Lix, “A Performance Comparison of Dynamic vs Static Load Balancing Policies in a Mainframe, Personal Computer Network Model”, Proceedings Of The 39th IEEE Conference on Decision & Control, 2000.

[9] Ali M Alakeel, “A Guide To Dynamic Load Balancing In Distributed Computer Systems”, International Journal of Computer Science and Network Security, Vol. 10 No. 6, June 2010.

[10] Z. Zhang, and X. Zhang, “A Load Balancing Mechanism Based on Ant Colony and Complex Network Theory in Open Cloud Computing Federation”, Proceedings of 2nd International Conference on Industrial Mechatronics and Automation ICIMA), Wuhan, China, May 2010, pages 240- 243.

[11] Nidhi Jain Kansal, Inderveer Chana, “Cloud Load Balancing Techniques: A Step Towards Green Computing”, IJCSI, Vol. 9, Issue 1, January 2012.

[12] R. X. T. and X. F. Z.. A Load Balancing Strategy Based on the Combination of Static and Dynamic, in

Database Technology and Applications (DBTA), 2010 2nd International Workshop (2010), pp. 1-4.

[13] H. Mehta, P. Kanungo, and M. Chandwani, “Decentralized content aware load balancing algorithm for distributed computing environments”, Proceedings of the International Conference Workshop on Emerging Trends in Technology (ICWET), February 2011, pages 370-375.

[14] A. M. Nakai, E. Madeira, and L. E. Buzato, “Load Balancing for Internet Distributed Services Using Limited Redirection Rates”, 5th IEEE Latin-American Symposium on Dependable Computing (LADC), 2011, pages 156-165.

[15] Y. Lua, Q. Xiea, G. Kliotb, A. Gellerb, J. R. Larusb, and A. Greenber, “Join-Idle-Queue: A novel load balancing algorithm for dynamically scalable web services”, An international Journal on Performance evaluation, In Press, Accepted Manuscript, Available online 3 August 2011.

[16] Xi. Liu, Lei. Pan, Chong-Jun. Wang, and Jun-Yuan. Xie, “A Lock-Free Solution for Load Balancing in Multi-Core Environment”, 3rd IEEE International Workshop on Intelligent Systems and Applications (ISA), 2011, pages 1-4.

[17] J. Hu, J. Gu, G. Sun, and T. Zhao, “A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment”, Third International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), 2010, pages 89-96.

[18] A. Bhadani, and S. Chaudhary, “Performance evaluation of web servers using central load balancing policy over virtual machines on cloud”, Proceedings of the Third Annual ACM
Bangalore Conference (COMPUTE), January 2010.

[19] H. Liu, S. Liu, X. Meng, C. Yang, and Y. Zhang, “LBVS: A Load Balancing Strategy for Virtual Storage”, International Conference on Service Sciences (ICSS), IEEE, 2010, pages 257-262.

[20] Y. Fang, F. Wang, and J. Ge, “A Task Scheduling Algorithm Based on Load Balancing in Cloud Computing”, Web Information Systems and Mining, Lecture Notes in Computer Science, Vol. 6318, 2010, pages 271-277.

[21] YashpalsinhJadeja, KiritModi, 2012 "Cloud Computing- Concepts, Architecture and Challenges"International Conference on Computing, Electronics and Electrical Technologies, IEEE, pp: 4/12.

[22] M. Randles, D. Lamb, and A. Taleb-Bendiab, “A Comparative Study into Distributed Load Balancing Algorithms for Cloud Computing”, Proceedings of 24th IEEE International Conference on Advanced Information Networking and Applications Workshops, Perth, Australia, April 2010, pages 551-556.

[23] http://www.loadbalancing.org/

[24] Z. Zhang, and X. Zhang, “A Load Balancing Mechanism Based on Ant Colony and Complex Network Theory in Open Cloud Computing Federation”, Proceedings of 2nd International Conference on Industrial Mechatronics and Automation (ICIMA), Wuhan, China, May 2010, pages 240- 243.

[25] S. Wang, K. Yan, W. Liao, and S. Wang, “Towards a Load Balancing in a Three-level Cloud Computing Network”, Proceedings of the 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT), Chengdu, China, September 2010, pages 108-113.

[26] V. Nae, R. Prodan, and T. Fahringer, “Cost-Efficient Hosting and Load Balancing of Massively Multiplayer Online Games”, Proceedings of the 11th IEEE/ACM International Conference on Grid Computing (Grid), IEEE Computer Society, October 2010, pages 9-17.

[27] R. Stanojevic, and R. Shorten, “Load balancing vs. distributed rate limiting: a unifying framework for cloud control”, Proceedings of IEEE ICC, Dresden, Germany, August 2009, pages 1-6.

[28] Y. Zhao, and W. Huang, “Adaptive Distributed Load Balancing Algorithm based on Live Migration of Virtual Machines in Cloud”, Proceedings of 5th IEEE International Joint Conference on INC, IMS and IDC, Seoul, Republic of Korea, August 2009, pages 170-175.

[29] A. Singh, M. Korupolu, and D. Mohapatra, “Server-storage virtualization: integration and load balancing in data centers”, Proceedings of the ACM/IEEE conference on Supercomputing (SC), November 2008.

[30] A Book by O’ Reilly on “Cloud Security And Privacy”.

[31] Sri Varsha Gorge, Virajith Jalaparti and Harini Vaidhyanathan “Multi-Tier Distributed Load Balancing” CS598RHC Literature Survey.

[32] A. Khiyaita, M. Zbakh, H. El Bakkali and Dafir El Kettani, “Load Balancing Cloud Computing: State of Art” , 9778-1- 4673-1053-6/12/$31.00, 2012 IEEE.