Volume 4, Issue 2, February 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Prediction Based Policies for Efficient Selection of Cloud Data Center

Sonali Rathod*, Urmila Shrawankar, Deepak Kapgate

Department of CSE. ,GHRAET,Nagpur

Nagpur University (MS), India

Abstract— Cloud computing has a variety of computing resources to facilitate the execution of large-scale tasks. Therefore, to select appropriate data center for executing a task is an important parameter to improve the performance of cloud computing environment. In present condition, available policies for data center selection are Round-robin, least connection, weighted least connection and so on. But they have some disadvantages like they did not focus on data centers load as well as weight has fixed value depending on the performance of data center. So we require more advance policies to improve the performance. In cloud environment, the response time observed at client side is mainly depends on load balancing algorithms and cloud data center selection technique. In this work we are proposing the data center selection prediction based techniques which predict the response time based on the history of response time for respective data center in the cloud and the selection of data center for upcoming request is done based on minimum predicted response time for each corresponding data center. Prediction based data center selection policy leads to effective reduction in response time observed at client side.

Keywords— cloud computing, data center selection policy, multiple variable regression forecasting model, prediction based algorithm.

I. INTRODUCTION

Since last few decades cloud computing becomes very popular. It provides number of options for everyday user as well as business application. Cloud computing refers to both the applications delivered as service over the Internet and the hardware and system software in the data center that provide those services. Cloud computing is a type of computing in which resources are shared. So, it allows user to use resources according to their needs. The whole internet can be viewed as a cloud. So, we can define cloud computing as internet based computing in which different services are provided to organizations computers. There are many existing issues in cloud computing like load balancing, virtual machine migration, energy management etc. The load balancing issue is the central issue in which the mechanism is distributing the dynamic workload to the nodes in whole cloud to achieve high user satisfaction and better resource utilization.

Before we reach existing load balancing algorithms ,we should focus on challenges of cloud computing. Challenges of Load Balancing In Cloud Computing are Performance - It is used to check the efficiency of the system, Throughput -It is used to calculate the number of task whose execution is completed, Resource Utilization -It is used to check utilization of resources [1], Scalability - It is the ability of load balancing algorithm to balance the load among larger number of nodes [1], Response Time - It is the time taken by the server for the particular request from the client.[1], Fault Tolerance - It is the ability of load balancing algorithm to tolerate the fault whenever any fault occurs.[2]

The main goal of a cloud-based architecture is to provide elasticity, & efficiently distribute the workloads or multiple requests to the particular data center. In cloud environment, the response time observed at client side is mainly depends on load balancing algorithms and cloud data center selection technique. Solution to above challenges can be use of effective algorithms for distributing the load among the nodes as well as selecting proper data center in cloud environment. Two types of algorithms are classified as static & dynamic. Static algorithms are suitable for homogeneous and stable environment. We cannot change any instance or attribute during execution time. Dynamic algorithms are more flexible we can add any instance or attribute during execution time. We will discuss these algorithms in next section.
Services are the applications provided by the servers across the cloud. Three types of services given in cloud.

A. Software as a service (SaaS)

In SaaS , the users uses application from different servers through internet. User can use this software application without any change. The client has to pay for particular time when he or she uses the software. The customer who don't want any software development but he is interested in high powered application then it can be benefited from SaaS. Some of the SaaS application are Video conferencing, Accounting, Web content management etc. The biggest benefit of SaaS is spending less money than buying the whole application.

B. Platform as a service (PaaS)

PaaS provides all types of resources which are required for constructing any application. There is no need of downloading or installing software. PaaS allows client to access a computing platform over a cloud computing solution. PaaS services include operating system, load balancer, scheduler, virtual machine management etc.

C. Infrastructure as a service (IaaS) or Hardware as a service (HaaS)

It provides the ways to manage or control the infrastructure of data centers in cloud. HaaS allows user to rent for resources such as server space, connecting infrastructure, memory, cpu , storage space. In this paper we present the survey of existing load balancing techniques and algorithms. The rest of the paper is organized as follows: Section II, focuses load balancing in cloud computing. Section III, discusses about the existing load balancing techniques in cloud computing. Section IV, discusses about load balancing algorithms. Section V and section VI shows comparative analysis of selection techniques and algorithms based on metrics. Finally, section VII concludes the paper.

II. LOAD BALANCING

Load balancing can be achieved by data center selection, virtual machine management, task scheduling. Load balancing means how to balance the client's requests across server i.e. how traffic is distributed in a cloud environment. The benefits of distributing requests include increase in resource utilization ratio, performance; therefore achieving maximum client satisfaction. Load balancing can be categorised in following areas.

A. Task Scheduling

A cloud computing environment where the millions of user are accessing thousand of servers all time. Scheduling the tasks/Jobs at the server is very difficult for a server, because the number Job requesting is very large in number, which required some resources to execute. The task schedule to be build by the server must be good so that each request by the user gets response in time, and every Task/Job gets proper resources for its execution.

B. VM Scheduling

The number of cloud users has been growing exponentially leads to effective scheduling of virtual machines. In cloud computing, a user may require a set of virtual machine co-operating with each other to accomplish one task. VMs need to be scheduled on the cloud in order to maximize utilization, Do the job faster, Consume less energy.

C. Data Center Selection

In cloud environment, the response time observed at client side is mainly depends on load balancing algorithms and cloud data center selection technique. Which data center should be selected for multiple incoming requests from the client is an important issue in cloud computing. In this context, Load balancing means how to balance the client's requests across server i.e. how traffic is distributed across data center in a cloud environment.

III. EXISTING LOAD BALANCING TECHNIQUES

In this section we discuss load balancing techniques with different literatures.

An artificial bee algorithm [3] which is an optimization method based on the gathering behavior of honeybees. Honeybee Foraging Behavior is one of the techniques inspired from the nature in which honeybee searches for the best nest site between many sites with taking care of both speed and accuracy. Bee uses waggle dance to communicate. This technique uses collection of servers arranged into virtual server. Each server has its own server queue. Each server is processing a request from its queue. Profit is calculated by each server serving a request from a queue-representative of the bee's measure of quality. One measure of this profit or reward can be the amount of time that the cpu spends on the processing of the request. This algorithm arranges a little link between requests in the same server queue, therefore improvement in system throughput.

The principle behind active clustering technique [4] is to group similar nodes together and then work on these groups. The process of creating cluster is the concept of matchmaker node. In this process initial node selects neighbor node called matchmaker node. The selection is based on the criteria that node should be of different type than the former one. This matchmaker node makes connection with its neighbour which is of same type. Finally the matchmaker node gets separated. This process followed iteratively. This technique provides efficient utilization of resources, high availability of resources therefore increase in throughput.

A join-idle queue [5] was presented for large scale load balancing with multiple dispatchers. At each dispatcher there is a queue, which stores a list of processors that have reported to be idle. Multiple requests or job comes to the dispatchers, and then the dispatcher removes the first idle processor from queue & directs the job to the idle processor. If the queue is empty, the dispatcher directs the job to the processor which is chosen randomly. When a processor becomes idle, it informs a queue of its idleness, or joins the queue. Each idle processor joins only one queue to avoid extra communication. This technique is basically used for large systems. It reduces load effectively and does not increase actual response time. Compare and Balance algorithm[6] allocates the requests of client to the lightly loaded server cluster or data center & gives the response in minimum time. In this technique allocation of requests is depend on the counter variable. Each data center has their counter value. The requests are sent to the data center whose counter value is minimum value. It focuses on fault tolerance to improve the performance.

An on- demand resource allocation and load balancing method for massively multiplayer online games is based on event driven technique[7]. An event is any identifiable occurrence that has significance for system hardware or software. Such events include both, user generated actions like mouse clicks and keystrokes. This type of feature is used in games. In event driven technique event is indicated as input, the components as a resources. This technique is normally used to balance the game sessions. Based on the predicted sessions and resource load in the next time interval, the resource allocation service arranges correct amount of resources for the proper execution. The main goal of resource allocation service is to extend game sessions to accommodate an increased number of players during peak hours. They design event driven load balancing solutions that receives capacity events from capacity planning service, directs the resource allocation service. Resource utilization is good. It has occasional QOS breaches. Fault tolerance is not focused.

Biased Random Sampling technique [8] is used for efficient use of resources in grid networks. Here a graph is constructed with connecting nodes. Nodes can be considered as servers or data centers. For data centers each with in-degree indicates that there is a free resource for particular data center. Whenever a node executes a job, it deletes an incoming edge which specifies reduction in the availability of free resource. After completing a job, the node creates incoming edge, which indicates an increase in the availability in the free resource. The addition and deletion of process is done by random sampling. Baised random sampling is the process where the node is randomly picked up with equal probability. The sampling start at fixed node, then it moves to neighbour node which is chosen randomly. The addition and deletion process of edges assures that load will be distributed equally across all nodes in network. The performance of the system is improved. Therefore increase in throughput due to the effective resource utilization.

A vector dot technique[9] was discussed to address the complexity of data center topology and multidimensional situations while deciding what items to move and where [10]. The author has tried to arrange virtual machine on physical machine so that they should balance the load. It uses NodeLoadFracVec which gives the load of node. This technique uses dot product of load fraction vector to choice a particular node for allocation of virtual machine from overloaded node. It provides effective resource utilization.

A scheme called CARTON [11] which solves the optimization problem. It has two concepts. First is the subgradient method to distribute the load to different data centers so that the cost can be minimized. The second concept is distributed rate limiting. DRL algorithm that allocates server capacities which ensures that performance levels at all data centers are equal.drl is used to make sure that resources are properly allocated or not. This is simple and easy technique to implement. Resources are good utilized.

A scheduling strategy on load balancing of VM resources[12] which uses historical data and current state of the system for load balancing. It reduces migration by using genetic algorithm, therefore solving issue of load balancing. It resolves issue of high cost of migration and better resource utilization.

Central load balancing for virtual machine(CLBVM) algorithm [13] balances the load over virtual machine as well as in cloud environment. This load balancing policy uses central dispatchers. This policy has centralized information and location rules. The transfer rule is partially distributed and partially centralized. Load balancing decision is based on global state information. This will improve the overall performance but not considered fault tolerant issue.

A load balancing virtual storage strategy [14] provides a large scale net data storage model and Storage as a Service model based on Cloud Storage. Storage virtualization is achieved by three layered architecture. Load is balanced by implementing two modules. It improves the efficiency, reduces the response time. It also improves flexibility, robustness of the system.

OLB (Opportunistic Load Balancing) and LBMM (Load Balance Min-Min) scheduling algorithm [15] has three levels, request manager, service manager, and service node. Opportunistic load balancing OLB is inherited with LBMM algorithm which can help in assigning task in free order to useful node. This algorithm solves the problem of min-min algorithm. LBMM scheduling algorithm is utilized to minimize the execution time of each task on the node therefore minimizing the overall completion time. This algorithm is used for efficient utilization of resources.

Task scheduling mechanism based on load balancing[16] first maps the tasks to virtual machine and then from virtual machine to host resources. This algorithm is used to achieve requirements of user. This improves the feature of task response time, overall performance, and resource utilization in cloud computing environment.

Workload and client aware policy (WCAP) algorithm[17] was implemented in decentralized manner. It uses the concept of USP, the unique and special property of request and computing nodes. For this it divides the web pages into core files and part files based on their access frequency. The core files are highly accessible which are associated with all nodes. The part files are divided among the nodes so that all nodes recieve equal number of file. The WCAP uses classification for the improvement in search process. By using the content information this technique improves the searching performance. Hence, It improves overall performance. It also helps in reducing the idle time of the computing nodes hence improving their utilization.

A new server-based load balancing policy for web servers [18] uses the protocol which limits the redirection of request to the closest server without overloading them. A middleware is described to implement this protocol. Any overloaded server replica can redirect requests to other replicas. The process is based on combining limited rates of request redirection and a heuristic that helps web servers to tolerate overload. It reduces the service response time.

Load balancing solution based on lock free multiprocessing [19] which avoids the use of shared memory . Multiple load balancing processes run on single load balancer. This algorithm uses shared memory and lock to maintain a user session. It is achieved by modified Linux kernel. By using this a new socket option named as sock level hash is added for listening socket and provide a lock free multiprocessing load balancing solution. After enabling sock level hash option, kernel forwards request from same client to same process by using hash function. It improves overall performance of load balancer by running multiple load-balancing processes in one load balancer.

IV. EXISTING LOAD BALANCING ALGORITHMS

Based on the system information algorithms can be classified into two types:

A. Static load balancing algorithm
B. Dynamic load balancing algorithm

Static load balancing algorithm

In this type of algorithm load is balanced among the server based on the previous performance statistics obtained [20].These algorithms are not depend on current state of system. The main goal of this type of algorithm is to reduce

overall execution time. Static algorithms are suitable for homogeneous and stable environment. We cannot change any instance or attribute during execution time. In this type of algorithm decisions are made at compile time. The advantage of this static algorithm is its simplicity in terms of both implementation and overhead. Round-robin algorithm, Randomized algorithm, Central load balancing algorithm fall under this group.

In Randomized algorithm [21] when client sends multiple requests to the server in cloud environment, server is randomly selected without any knowledge of whether that particular server has heavy load or minimum workload. Randomized algorithm works well with infinite number of processes. It provides best performance among all load balancing algorithms for special purpose applications.

Round-robin [22] distributes client requests across multiple servers or data center in cloud environment. Round robin passes each new request to the next data center. Round robin algorithm sends request to each data center even it has heavily loaded or idle. Advantage of this algorithm is it is simple and cheap and very predictable. The problem discussed above can be reduced by using central load balancing (CLBDM) which is one of the static load balancing algorithm. Round-robin distributes request equally, which produces lots of problem. Hence weighted round-robin comes under consideration.

Weighted Round-robin is the extension of basic round-robin algorithm with weights. Weighted Round-robin assigns the request to particular data center as same as their weight and then transfers remaining requests to other data centers according to their weights. That means data centers will receive a number of request in proportion to their weight. Weighted round robin[23] algorithm assigns more requests to nodes with a greater weight. Two methods were presented based on hierarchical structure. First method determines weight based on state information of system . second method Minimum Load State Round Robin(MLSRR) uses state information and finds the nodes with minimum load and transmits tasks to these nodes.

One algorithm (CLBDM)[24] which was tackled the problem of overloaded server in round-robin algorithm. They proposed central load balancing decision model in which interaction with all parts in a system is done. By using this information CLBDM follows the process. For this they include user experience sensors which has user sessions like all request sent and received by user, and time required to complete the action. If response time rises CLBDM passes the load to other servers according to the round robin algorithm.

Even ants have very limited memory; they manage to perform a variety of complicated task with reliability & consistency. Ant colony optimization technique [25] has the ability to find the shortest route inspiring from ants behaviour. In the case of load balancing in cloud environment, as the web server demands increases or decreases, the Services are assigned dynamically to regulate the changing demand of the user. In this technique routing table is constructed for an entry of each node .The table is called as pheromone table. Each row in the table represents the routing preference for each destination & each column represents the probability of choosing a neighbour as the next hop. Server has its own queue. Server serves request, calculates profit & compare it with colony profit & then set the probability. If the profit is high, then stays at current server. If it is low then server return to the table. Ants moves from node to node, selects next node according to probabilities in the table for their destination node. At the arrived node they update the probabilities of that node's table entries corresponding to their source node. The pheromone update mechanism has an efficient and effective tool to balance the load. This technique does not consider the fault tolerance issues.

The model of reducing tasks was presented which works in cluster environment with thousands of nodes. Mapreduce [26] program execution has two main functions, Map and Reduce. These functions can execute in parallel. The Map function is called for each input key value and the Reduce function is called for each key from the Map output. Author proposed Blocksplit load balancing algorithm in which for mapping task, the request is partitioned into parts, and then similar entities are grouped together. Mapping task processes in parallel with reduce task, hence reduce the overloading.

The algorithm for mapping virtual machine to physical machine [27] contains central scheduling controller and resource monitor. The resource monitor maintains information about available resources. And scheduling controller assigns the task to the particular resource. In this algorithm virtual machine accepted request first. Then find the information of available resource using resource monitor. Now controller calculates the ability of resource to handle the task. Finally, highly scored resource is selected to handle the task and client will access its application.

Dynamic Load Balancing Algorithms

Dynamic load balancing algorithm is not based on prior knowledge of system, it is depend on current state of system [20].Dynamic algorithms are more flexible we can add any instance or attribute during execution time. It provides better result in heterogeneous & dynamic environments. However, as the distribution attribute become more complex and dynamic. As a result some of these algorithms could become inefficient and cause more overhead than necessary resulting in an overall degradation of the services performance. Least connection, weighted least connection, join idle queue algorithm etc. comes under this group. Least connection algorithm[28] for load balancing sends requests to data center based on which data center is currently having fewest connections. Total number of connection is identified at run time. However, least connection does not consider distance between client and server and other factors.

Weighted least connection algorithm[29] maintains a weighted list of servers with number of connections and forward a new connection to the server based on its weights and number of connection. Weighted least connection algorithm specifies that the next data center is chosen for a new connection with a few connections. This method works best where the servers have different capacities. Weighted least connection algorithm maintains a weighted list of servers with the no of connection and forward a new connection to the server based on its weight and number of connections.

The weighted least connection algorithm specifies that the next data center is chosen for a new connection with the fewest connections. The model Exponential Smooth Forecast- based on weighted least connection(ESBWLC)was presented [30] in which selection of data center is depend on parameters like cpu power ,memory, performance etc. Finally it decides to which data center the load will be assigned by using exponential smoothing model. But this model has limitations that it does not consider the distance between data center and client which leads to negletion of communication delays and bandwidth available for sending request, which is over made by algorithm called Extended ESBWLC [31]. This research shows that how the selection of data center based on predicted response time of available data centers leads to minimization of load on data centers and reduction in latency felt by users.

An algorithm called Dual Direction algorithm for FTP (DDFTP)[32] have two different servers, and each server works concurrently, processing independently. DDFTP server moves the file from left to right while second server starts moving from right to left. The parallel DDFTP process stops as soon as they meet together. That means when client receives two blocks from each server it specifies that they both completed their work. The goal of this technique is that the fast DDFTP server connected to DDFTP client has opportunity to transfer more blocks. While the slow DDFTP server transfers fewer blocks. Hence achieving load balancing .DDFTP parallelizes downloads from multiple replicas to the client. This algorithm reduces communication overhead.

Load Balance Min-Min(LBMM) algorithm uses concept of min-min algorithm. The big disadvantage of min-min algorithm is it only focuses on the computation time for the task.[33] It did not focus on workload of the particular node. LBMM algorithm [34] solves the problem of min-min algorithm. This algorithm has three levels. Third level is service node which is used to execute subtask. Second level is service manager which divides task into subtasks. First level is request manager which is used to assign a task to service manager. Opportunistic load balancing OLB is inherited with LBMM algorithm which can help in assigning task in free order to useful node. LBMM distributes task among service manager, which considers execution time required for the subtask on each service node. Service manager chooses service node of minimum execution time to execute different subtask and maintain the record in mintime - array. That means sub task on service node performed first which required min execution time. Subtask will remove from queue after completion and mintime-array is arranged newly.

The algorithm Index Name Server [35] is used to reduce redundancy. INS helps in selecting optimum point Some of the parameters can be hash code of block of data to be downloaded, the position of server which has target block of data .some calculations are there to find out whether the connection is capable of handling additional nodes. They are classified as busy levels such as B (a),B (b),B(c). B (a) means that connection is very busy and cannot handle any additional connection. B (b) means connections is not busy and can handle additional connections. B(c) means that the connection is limited.

V. COMPARATIVE ANALYSIS OF SELECTION TECHNIQUES

TABLE I

Parameter	Performance	Response Time	Scalability	Overhead	Throughput	Resource Utilization	FaultTolerance
HoneybeeForagingBehaviour	YES	NO	YES	NO	YES[3]	NO	NO
Ant Colony Optimization	YES	NO	YES	NO	NO	YES	YES
Active Clustering	YES	NO	YES	NO	YES[4]	YES[4]	NO
Join-Idle Queue	YES	YES[5]	NO	YES	NO	NO	NO
Compare & Balance	NO[6]	NO[6]	NO	YES	NO	YES	NO[6]
Event Driven	NO	NO	YES	NO	NO	YES[7]	NO[7]
Baised Random Sampling	YES[8]	NO	YES	NO	YES[8]	NO[8]	NO
Vector Dot	NO	NO	NO	NO	NO	YES[9]	NO
CARTON	YES	NO	NO	YES	NO	YES[11]	NO
CLBVM	YES[13]	YES	NO	NO	YES	YES	NO[13]
LBVS	YES[14]	YES[14]	YES	NO	NO	NO	YES
OLB+LBMM	YES[15]	NO	NO	NO	NO	NO[15]	NO
TaskScheduling	YES[16]	YES[16]	NO	NO	NO	YES[16]	NO

Decentralized Content Aware	YES[17]	YES	YES	YES	NO	YES[17]	NO
Server based LB for internet distributed services	YES	YES	NO	NO	NO	NO	NO
Lock free multiprocessing solution for LB	YES	NO	NO	NO	YES	NO	NO

VI. COMPARATIVE ANALYSIS OF SELECTION ALGORITHMS

TABLE II

Parameters	**Performance**	**ResponseTime**	**Scalability**	**Overhead**	**Fault Tolerance**
CLBDM	NO	YES[24]	YES	YES	NO
Ant ColonyOptimization	YES	NO	YES	NO	NO[25]
MapReduced-Based Entity Resolution	YES[26]	NO	NO	YES	YES
VirtualMachine mapping	YES	NO	NO	YES	YES
ESBWLC	YES[30]	YES	NO	YES[30]	YES
LBMM	YES[34]	NO	NO	NO	NO
Index Name Server	YES	NO	NO	YES	NO

As per above comparison some of the techniques and algorithms are good for small scale area and some are good for large scale area. The selection of appropriate algorithm is depending on the customer's requirement. If customer needs better performance for their cloud environment then choice could be honeybee foraging, ant colony, active clustering, CLBVM, OLB+LBMM, ESBWLC, Map reduce entity resolution. If the requirement is high throughput and scalability then baised random sampling is good. For better resource utilization compare and balance, vector dot, event driven, CARTON, CLBDM, ESBWLC can be good choice.

VII. CONCLUSION

The response time and data transfer cost is a challenge of every cloud engineer that can increase the business performance in the cloud industry. Several strategies lack efficient scheduling and load balancing resource allocation techniques which leads to increase operational cost and customer dissatisfaction. Response Time is the amount of time taken to respond by a particular data center in a cloud system. This parameter should be minimized. In this paper several techniques and algorithms are discussed from which any algorithm can used according to the customer requirement. Some of the techniques and algorithms are good for small scale area and some are good for large scale area. According to the comparison some techniques have limitations so they could not select data center properly. Due to this performance is degraded. So there is need to develop algorithm for heterogeneous environment as well as to select data center appropriately which should satisfy customers requirement. Prediction based data center selection policy leads to effective selection of data center in cloud computing environment.

REFERENCES

[1] Buyya R., R. Ranjan and RN. Calheiros, "Inter Cloud: Utility oriented federation of cloud computing environments for scaling of application services," in proc. 10th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP), 2010.

[2] Rimal, B. Prasad, et al, "A taxonomy and survey of cloud computing systems." In proc. 5th International Joint Conference on INC, IMS and IDC, IEEE, 2009.

[3] Jing Yao et al ," Load Balancing Strategy of Cloud Computing based on Artificial Bee Algorithm"8thInternational Conference on Computing Technology and Information Management (ICCM), IEEE ,Volume:1 ,April 2012 .

[4] Pragati Priyadarshinee et al, "Load Balancing and Parallelism in Cloud computing" International Journal of Engineering and Advanced Technology (IJEAT), ISSN: 2249 – 8958, Volume-1, Issue-5, June 2012.

[5] Yi Lu, et al," Join-Idle-Queue: A Novel Load Balancing Algorithm for Dynamically Scalable Web Services" The 29th International Symposium on Computer Performance, Modeling, Measurements and Evaluation, 2011.

[6] shreyas mulay et al"enhanced equally distributed load balancing algorithm for cloud computing" international journal of research in engineering and technology, vol no:02, issue:06, june 2013.

[7] Nae, V.; Prodan, R.; et al , "Cost-efficient hosting and load balancing of Massively Multiplayer Online Games," 11th IEEE/ACM International Conference on Grid Computing (GRID), pages 9-16, 25-28 Oct 2010.

[8] O. Abu- Rahmeh, P. Johnson et al, "A Dynamic Biased Random Sampling Scheme for Scalable and Reliable Grid Networks", INFOCOMP - Journal of Computer Science, ISSN 1807-4545, VOL.7, pages 01-10,December2008.

[9] Singh A et al," Server-storage virtualization: Integration and load balancing in data center" International Conference On High Performance Computing, Networking, Storage and Analysis, IEEE, pages 1-12, nov 2008.

[10] Nidhi Jain Kansal1 et al," Cloud Load Balancing Techniques : A Step Towards Green Computing" IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 1, January 2012.

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

[12] Hu J., Gu J., et al, "A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment"3rd International Symposium on Parallel Architectures, Algorithms and Pro-gramming, IEEE,89-96, 2010.

[13] Bhadani A. and Chaudhary S. (2010) "Performance evaluation of web servers using central load balancing policy over virtual machines on cloud" COMPUTE 10 proceedings of the third annual ACM Bangalore Conference, 2010.

[14] Liu H., Liu S., et al," LBVS: A Load Balancing Strategy for Virtual Storage" IEEE International Conference on Service Sciences (ICSS), 257-262, May 2010.

[15] Wang, S-C., K-Q. Yan,et al, "Towards a load balancing in a three-level cloud computing network," in proc. 3rd International Conference on. Computer Science and Information Technology (ICCSIT), IEEE, Vol. 1, pp: 108-113, July2010.

[16] Fang Y., Wang F. et al, Lecture Notes in Computer Science, 6318, 271-277,2010.

[17] H. Mehta, P. Kanungo, M. Chandwani, "Decentralized content aware load balancing algorithm for distributed computing environment," Accepted in International Conference and Workshop on Emerging Trends and Technology, Mumbai, Feb 2011.

[18] Nakai A.M., Madeira E,et al, "Load Balancing for Internet Distributed Services Using Limited Redirection Rates" 5th Latin-American Symposium on Dependable Computing, IEEE,156-165,2011.

[19] Liu Xi., Pan Lei.,et al,"Xi Liu; Lei Pan; Chong-Jun Wang; Jun-Yuan Xie (2011), A Lock-Free Solution for Load Balancing in Multi-core Environment" 3rd International Workshop on Intelligent Systems and Applications, IEEE,1-4,2011.

[20] Sairam Vakkalanka ," A Classification of Job Scheduling Algorithms for Balancing Load on Web Servers" International Journal of Modern Engineering Research (IJMER) , Vol.2, Issue.5, Sep-Oct. 2012 .

[21] R. Motwani and P. Raghavan, "Randomized algorithms", ACM Computing Surveys (CSUR), 28(1):33-37, 1996.

[22] Sotomayor, B., RS. Montero, et al, "Virtual infrastructure management in private and hybrid clouds," in IEEE Internet Computing, Vol. 13, No. 5, pp: 14-22, 2009.

[23] Iman Barazandeh et al, "Two New Biasing Load Balancing Algorithms in Distributed Systems",IEEE,2009.

[24] Radojevic, B. and M. Zagar, "Analysis of issues with load balancing algorithms in hosted (cloud) environments." .34th IEEE International Convention on MIPRO, 2011.

[25] Ratan Mishra et al," Ant colony Optimization: A Solution of Load balancing in cloud" International Journal of Web & Semantic Technology (IJWesT) Vol.3, No.2, April2012

[26] Kolb, L., A. Thor, and E. Rahm, E, "Load Balancing for MapReduce-based Entity Resolution," 28th IEEE International Conference on Data Engineering (ICDE), pp: 618-629, 2012

[27] Ni, J., Y. Huang, Z. Luan, et al, "Virtual machine mapping policy based on load balancing in private cloud environment," in proc. International Conference on Cloud and Service Computing (CSC), IEEE, pp: 292-295, December 2011.

[28] Tian Shaoliang, Zuo Ming and Wu Shaowei, " An improved load balancing algorithm based on dynamic feedback", Computer Engineering and Design, 28: 572-573,2007.

[29] Zheng Qi.,et al," Load balancing algorithm based on dynamic feedback", Computer Age, pp: 49-51,2006.

[30] Ren, X., R. Lin and H. Zou, "A dynamic load balancing strategy for cloud computing platform based on exponential smoothing forecast" International Conference on. Cloud Computing and Intelligent Systems (CCIS), IEEE, pp: 220-224, September 2011.

[31] D.Kapgate, et al "Predictive Load Balancing Strategy for reduction of Latency in Mobile Cloud Computing" International Journal of Computer & Communication Engineering Research (IJCCER) Volume 1 - Issue 1 May 2013.

[32] Al-Jaroodi, J. and N. Mohamed. "DDFTP: Dual-Direction FTP," in proc. 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid), IEEE, pp: 504-503, May 2011.

[33] Dai You et al,"Reasearch On Resource Scheduling Algorithm Based On ForCES Network" Information Technology Journal 12(12):2419-2425, 2013.

[34] Shu-Ching Wang, Kuo-Qin Yan," Towards a Load Balancing in a Three-level Cloud Computing Network"IEEE,2010.

[35] T-Y., W-T. Lee, et al, "Dynamic load balancing mechanism based on cloud storage" in proc. Computing, Communications and Applications Conference (Com Com Ap), IEEE, pp: 102-106, January 2012.

