Volume 5, Issue 2, April 2018 ISSN: 2348-8565 (Online)
International Journal of Modern Engineering and Research Technology
Website: http://www.ijmert.org Email: editor.ijmert@gmail.com

Load Balancing Algorithms in Cloud Computing-A Survey

Loveleen Kaur
Assistant Professor
Jabalpur Engineering college Jabalpur
Jabalpur, (M.P.) [INDIA]
Email: lpabla@jecjabalpur.ac.in

ABSTRACT

Different models and algorithms for load balancing in cloud computing has been developed with the aim to make cloud resources accessible to the end users with ease and convenience. In this paper, we aim to provide a planned and wide-ranging summary of the research on load balancing algorithms in cloud computing. This paper surveys load balancing tools and techniques over the period of 2012-2016. We are providing collection of existing approaches aimed at providing load balancing in a reasonable manner. With this classification we provide an easy and brief view of the underlying model adopted by each approach.

Keywords:—*cloud computing; load balancing; algorithms, load balancers;*

1. INTRODUCTION

Cloud computing provides flexible way to retain data and files which involves virtualization, distributed computing, and web services. It also has several elements like client and distributed servers. The aim of cloud computing is to provide maximum services with minimum cost at any time. Nowadays, there are more than hundred millions of computer devices connected to the Internet. These devices submit their request and receive the response without any delay. Figure 1, shows different devices (tablet, PCs, laptops) connect and access the data from a cloud at any given time. The main objectives of cloud are to reduce cost, enhance response time, provide better performance, hence Cloud is also called a pool of services [1]. Load has various types like, CPU load, network load, memory capacity issue etc. In the context of cloud computing, load balancing is to share load of virtual machines across all nodes (end user devices) to improve resources, service utilization and provides high satisfaction to users. Due to load sharing, every node can work efficiently, data can be received and sent without delay [2]. The dynamic load balancing [3] algorithm uses system information while distributing the load. A dynamic scheme is more flexible and fault tolerant. Load balancing enables advance network facilities and resources for better response and performance. Several algorithms are used to balance cloud data among nodes. All the user load is handled by cloud provider for smooth provisioning of services. Therefore, the proposed algorithm will be used by cloud service provider (CSP).

Load balancing is usually applied on huge amount of data traffic and servers to distribute work. Advanced architectures in cloud are adopted to achieve speed and efficiency. There are several characteristics of load balancing such as: equal division of work across all the nodes, facilitation in achieving user satisfaction, improve overall performance of system, reduce response time, and provide services to achieve complete resource

utilization [4]. Figure 2 shows the load balancing in cloud computing. As an example, if we make one application on cloud and hundreds of users are expected to access it at any one time. Therefore, response time to hundred people will be very slow and servers will become busy very quickly, resulting in slow response and unsatisfactory users. If we apply load balancing on our application, then work will be distributed at other nodes and we can get high performance and better response [5]. The existing survey does not critically discuss the available tools and techniques that are used in cloud computing.

In this paper, we provide a comprehensive overview of interactive load balancing algorithms in cloud computing. Each algorithm addresses different problems from different aspects and provides different solutions. Some limitations of existing algorithms are performance issue, larger processing time, starvation and limited to the environment where load variations are few etc. A good load balancing algorithm should avoid the over loading of one node. The aim is to evaluate the performance of the cloud computing load balancing algorithms that have been developed over the period of 2004-2015. The rest of the paper is organized as follows. In section II, we compare review different load balancing algorithms. In Section III, the performance evaluation of different cloud computing algorithms have been discussed and evaluated with the help of multiple tables. Our discussion and findings are summarized and the paper is concluded in section IV.

Figure 1. A Cloud Computing Scenario [7]

II. Related Work

The process in which the load is divided among several nodes of distributed system is called load balancing in cloud computing [6]. Load balancing assists the cloud computing through algorithms [7]. Lots of work has been done to balance the load in order to improve performance and avoid over utilization of resources. Various load balancing algorithms have been discussed including round robin (RR), Min-Min, Max-Min etc. Load balancing algorithm are divided in two main categories, namely static and dynamic [33]. Figure 3 shows the classification of load balancing algorithms. In this section, we provide a detailed discussion on the existing load balancing algorithms for cloud.

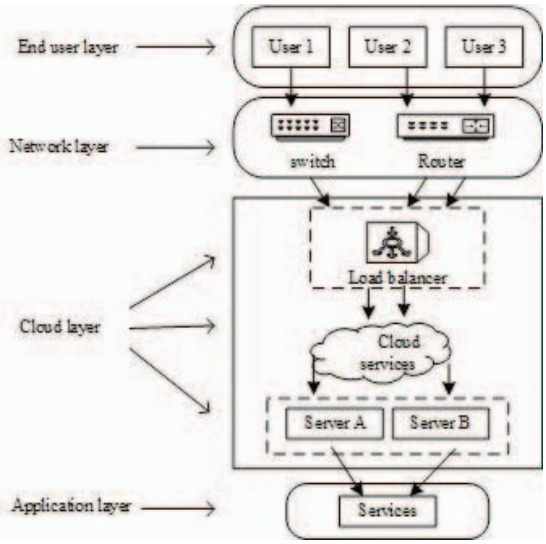

Figure 2. Load balancing in cloud computing

A. Static Algorithm

These algorithms are based on completion time of a task [8]. In static algorithms decision about load balancing is made at compile time. These are limited to the environment where load variations are few. These algorithms are not dependent upon the present condition of system. A static load balancer algorithm divides the traffic equally among the servers. It does not use the system information while distributing the load and is

less complex [9].A specific weight is assigned to the server. Server having highest weight receives more connections comparatively. Job is assigned according to the ability of the node. In static algorithm, dynamic changes at run-time are commonly not deliberated. Along with this, static algorithm do not have the ability to handle load changes throughout run-time [10]. Radojevic proposed CLBDM (central load balancing decision model) and this is an advance form of Round Robin algorithm [11]. This algorithm works properly in a system with low variation of load [12]. In CLBDM connection time between client and node is calculated [13]. This algorithm can be unreliable because of unforeseen loops. Static algorithm transfers only fix amount of data [14][15]. It has no ability to fault tolerance [16].

1) Round Robin Load Balancer

In round robin fixed quantum time is given to the job. Main emphasis in round robin is on fairness and time limitation. It uses the ring to line up the collected tasks. It utilizes equal period to complete each task [17] [18] [19]. In case of heavy load, round robin takes a long term to finish all the given tasks. In case of larger tasks it takes longer time for completion [20]. In round robin loads are equally distributed to all VMs. Few limitations of this algorithm suggest that, to achieve high performance, more than one client connections should not start at the same time [21].As the name round robin shows, it works in a circular pattern. Each node is fixed with a time slice and performs a task at designated time on its turn. It is less complex [22][23] [24]. As a result, at any moment some node may possess heavy load and others may have no request. Therefore, it is not useful for cloud computing [25][26][27][28]. This problem was tackled by weighted round robin [29] where each node is allowed to receive specific number of requests according to the assigned weight [30][31][32].

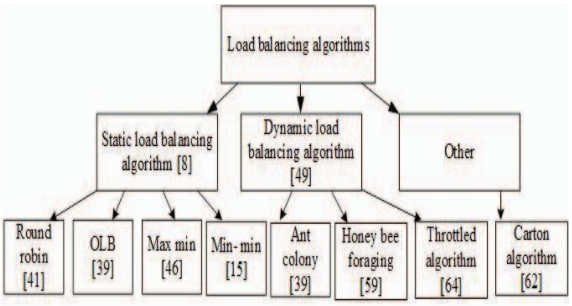

Figure 3. Classification of load balancing algorithms.

2) OLB + LBMM

Wang *et al* [39], suggested a combination of opportunistic load balancing (OLB) and load balancing min-min (LBMM) algorithms to improve the performance of tasks. By this algorithm, resources can be used more effectively and it increases the task proficiency. All tasks are given to the nodes in a specific manner. Its results are better than all other algorithms [40] and it is used in LBMM. LBMM works in three levels: in the first level it behaves like a request manager. It manages to receive and allocate the task to service manager. When the request is received by service manager, the task is divided into chunks to speed up the process. After that it allocates these chunks to the node. The assigning of tasks is based on available nodes, cpu capacity and remaining memory [38]. Task completion and running time of the node is not considered in OLB. That is why the tasks take much time for completion. Often requests are found in waiting list, till nodes become free to take other tasks [41][42].

3) Max - Min

Max-min [46] is same as min-min algorithm. But max-min chooses the task with maximum value and gives to the respective machine. After assign the task, machine works according to updates. These assigned tasks remove from the list [47]. The chosen node and tasks arrange in a specific pattern updates about the ready time are given by combining the running time of the job [48].

Table 1. Merits and Demerits of Load Balancing Algorithms.

Scheduling Algorithms	Merits	Demerits
Static load balancing	Decision about load balancing is made at compile time. Divides the traffic equally among the server. Fewer complexes.	Limited to the environment where load variations are few. Do not have ability to handle load changes throughput runtime.
Round Robin	Fixed time quantum.; Easy to understand; Fairness Performs better for short CPU burst. Also used priority (running time and arrival time).	Larger tasks take long time. Can occur more context switches due to short quantum time Job should be same to achieve high performance.
Min- Min	Smallest completion time value. In presence of more small tasks, it shows best result	Starvation Machine and tasks variation can't be predicted.
Max – Min	Requirements are prior known. So works better.	It takes long time to complete the task.
Dynamic load balancing	Distribute work at run time; Fault tolerance Only current state of system is required.	Need constant check of the nodes. Considered more complicated.
Honey Bee	Increases throughput; Minimize response time.	High priority tasks can't work without VM machine.
Ant - Colony	Faster information can be collected by the ants.; Minimizes make span.; Independent tasks; Computationally intensive	Network is over headed so search takes long time. No clarity about the number of ants.
Carton	Fairness; Good performance; Equal distribution of responses. Low communication is required.	It depends upon lower costs.
Throttled load balancing	Good performance; List is used to manage the tasks.	Tasks need to be waited.

B. Dynamic Load Balancer

Dynamic algorithm is based on the different properties of the nodes such as capabilities and network bandwidth. This need constant check of the node and are usually difficult to implement [49][50]. Dynamic algorithms are well suited in cloud computing environment because they distribute work at run time and assign suitable weights to the servers. A lightest weight server is search in network and preferred by this algorithm [51]. Dynamic algorithms are consider more complicated [52]. Ran [53] suggested WLC [53] (weight least connections), a dynamic load balancing algorithm for cloud computing. The WLC assign tasks on the basis of number of connections for existing node. In dynamic load balancing the load distributes among the nodes throughout run-time. If load balancer finds high usage of CPU the request is send to the next node [54]. To handle the load, current state of the system is used [55]. In dynamic load balancing, documents and data can be downloaded with at any restriction of particular memory. [56]. Its advantage comes forward, when any node is failed. In such situation, it does not stop the system, only its performance is affected [57].

1) Ant colony algorithm

Different ant colony algorithms also introduce to balance the load applying ant behavior for searching food. Larger weight means that resource has high computation power [61]. Load balancing ant colony optimization (LBACO) not only balance the load but also minimizes make span. All tasks are assumed to be mutually independent and computationally intensive.

2) Honey bee foraging algorithm

Dhinesh *et al* proposed an algorithm after detail analysis of foraging behavior of honey bees [58]. When an under loaded VM assigns a task, it updates number of priority tasks and

load of VM to other tasks in waiting list. This approach helps other processes to choose their VM [59]. If a task has high priority, then it selects a VM having minimum number of priority tasks. It does not take into consideration only load balancing but also keeps track of priorities of tasks which currently removed from heavy loaded machines [60]. It increases throughput and minimizes response time.

3) Throttled load balancing

This algorithm depends upon the theory of suitable search of virtual machine. The task manager makes a list of virtual machines. By using the list, client request allotted to the relevant machine. If the size and capability of the machine is suitable for request, then the job is given to that machine. This algorithm is better than round robin algorithm [59][64].

4) Carton

Carton [62] is a technique that is combination of load balancing (LB) and distributed rate limiting (DRL). Through LB, jobs are fairly assign to the servers. While DRL ensures the equal distribution of resources. Work load is dynamically assign to improve the performance and spread the load equally to all the servers. This algorithm can easily be implemented as low communication required [63].

III. PERFORMANCE EVALUATION

Table 1 describes the evaluation of the discussed Load Balancing (LB) algorithms through different parameters like fairness, throughput, waiting Time. In table 2, the comparison of these algorithms show positive and negative results and we describe this as high and low term. As discussed pervious different algorithms show different results. Such that, Static algorithm consider fair to distribute the load. But it is less complex and not fault tolerant. Min-Min algorithm is not fair and fault tolerant. In case of small tasks, it shows best result. In Max-Min, requirements are prior known. So it works better and gives high throughput. Along with this, dynamic

Table 2. Load Balancing Algorithms Criteria

LB algorithms	Fairness	Response time	Through-put	Overhead	Fault tolerance	Performance	Resource utilization	Speed	Complexity
Static [25] [51]	Yes	Fast	High	N/A	No	Fast	High	Fast	Low
Round Robin [41]	Yes	Fast	High	High	No	Fast	High	N/A	Low
Min-Min [15][9]	No	Fast	High	High	No	Fast	High	Fast	Low
Max-Min [44]	No	Fast	High	High	No	Fast	High	Slow	Low
Dynamic [23][11]	No	Slow	High	High	Yes	Slow	High	Fast	High
Honey bee [59]	No	Slow	High	Low	No	Slow	High	Fast	Low
Ant colony [27] [1]	No	Slow	High	High	N/A	Slow	High	Fast	No
Carton [62]	Yes	Fast	High	N/A	N/A	Fast	High	Fast	High
Throttle [32]	No	Fast	High	N/A	N/A	Fast	High	Fast	Low
OLB+ LBMM [39]	No	Slow	High	Low	No	Fast	High	Slow	High

load balancing requires only current state of the system and has more overhead and fault tolerance. Honey bee has high throughput and low response time. It has low overhead and performance because high priority tasks can't work without VM machine. Ant colony is simple algorithm and less complex. Carton algorithm requires low communication and its working is fair. Table 2 provides a detailed comparison of different algorithms over different parameters like fairness, performance, speed, complexity. We recommend that, Round Robin is more efficient according to following facts, Round Robin consider fair to distribute the load, it has high throughput, good response time and less complex than other algorithms. The major advantage of RR is time limitation and use equal period to complete each task.

4. CONCLUSION

In this paper, we have presented comparison of different load balancing algorithms for cloud computing such as, round robin (RR), Min-Min, Max-Min, Ant colony, Carton, Honey bee etc. We described advantages and limitations for these algorithms showing results in different conditions. The vital part of this paper is comparison of different algorithms considering the characteristics like fairness, throughput, fault tolerance, overhead, performance, and response time and resource utilization. The limitation of existing work is that each cloud computing algorithm does not address the related issues like fairness, high throughput and equality. Future work is to mitigate the above problem, and use the hybrid approach to attain better performance and secure the system.

REFERENCES:

[1] K. Li, G. Xu, G. Zhao, Y. Dong, and D. Wang, "Cloud task scheduling based on load balancing ant colony optimization," in Sixth Annu. Chinagrid Conf, pp. 3–9, Aug. 2011.

[2] A. Abraham, "Genetic algorithm based schedulers for grid computing systems Javier Carretero, Fatos Xhafa," in International Journal of Innovative Computing, Information and Control, vol. 3, no. 6, pp. 1–19, 2007.

[3] M. Katyal and A. Mishra, "A comparative study of load balancing algorithms in cloud computing environment." in International Journal of Distributed and Cloud Computing, Volume 1 Issue 2 December 2013.

[4] R. G. Rajan and V. Jeyakrishnan, "A survey on load balancing in cloud computing environments," in International Journal of Advanced Research in Computer and Communication Engineering, vol. 2, no. 12, pp. 4726–4728, 2013.

[5] X. Xu, H. Yu, and X. Cong, "A qos-constrained resource allocation game in federated cloud," Seventh Int. Conf. Innov. Mob. Internet Serv. Ubiquitous Comput., pp. 268–275, Jul. 2013.

[6] B. Godfrey, K. Lakshminarayanan, S. Surana, R. Karp, I. Stoica, "Load balancing in dynamic structured P2P systems", in INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, vol. 4, IEEE, 2004,

[7] K. Bala, S. Vashist, R. Singh, and G. Singh, "A review of the load balancing techniques," International Journal of Advances in Computer Science and Communication Engineering (IJACSCE), vol. 2, no. I, march 2014.

[8] H. Chang and X. Rong, "Legendre transform-dual solution for a class of investment and consumption problems with hara utility," Mathematical Problems in Engineering, vol. 2014, 2014.

[9] S. S. Chauhan and R. C. Joshi, "A weighted mean time min-min max-min selective scheduling strategy for independent tasks on grid," Advance Computing patiala Conference (IACC), 2010 IEEE 2nd International, pp. 4–9, 2010.

[10] I. Foster, "Virtual infrastructure management in private and hybrid clouds." in Internet Computing, IEEE,. Volume: 13, Issue: 5, 2009,

[11] N. Haryani and D. Jagli, "Dynamic method for load balancing in cloud computing," IOSR Journal of Computer Engineering (IOSR-JCE), vol. 16, no. 4, pp. 23–28, 2014.

[12] H. Izakian, A. Abraham, and V. Snasel, "Comparison of heuristics for scheduling independent tasks on heterogeneous distributed environments," Int. Jt. Conf. Comput. Sci. Optim., pp. 8–12, Apr. 2009.

[13] B. R. Kandukuri, R. P. V., and A. Rakshit, "Cloud security issues," IEEE Int. Conf. Serv. Comput., pp. 517–520, 2009.

[14] N. J. Kansal and I. Chana, "Cloud load balancing techniques : a step towards green computing," IJCSI International Journal of Computer Science issue, vol. 9, no. 1, pp. 238–246, 2012.

[15] H. Chen and F. Wang, "User-priority guided min-min scheduling algorithm for load balancing in cloud computing."Parallel Computing Technologies (PARCOMPTECH), National Conference IEEE, 2013

[16] M. T. Cse, "Comparison of load balancing algorithms in a Cloud Jaspreet kaur," International Journal on Cloud Computing: Services and Architecture (IJCCSA), vol. 2, no. 3, pp. 1169–1173, 2012.

[17] N. Sran and N. Kaur, "Comparative analysis of existing load balancing techniques in cloud computing," International Journal of Engineering Science Invention, vol. 2, no. 1, pp. 60–63, 2013.

[18] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," Journal of network and computer applications, vol. 34, no. 1, pp. 1–11, Jan. 2011.

[19] N. Swarnkar, "A survey of load balancing techniques in cloud computing," International Journal of Engineering Research & Technology, vol. 2, issue 8, pp. 800–804, 2013.

[20] N. J. Kansal and I. Chana, "existing load balancing techniques in cloud computing : a systematic re- view," Journal of Information Systems and Communication, vol. 3, no. 1, pp. 87–91, 2012.

[21] R. Kaur and P. Luthra, "Load balancing in cloud computing," Int. Conf. on Recent Trends in Information, Telecommunication and Computing, ITC, pp. 1–8, 2014.

[22] Z. Chaczko, V. Mahadevan, S. Aslanzadeh, and C. Mcdermid, "Availability and load balancing in cloud computing," --International

Proceedings of Computer Science and Information Technology, vol. 14, pp. 134–140, 2011.

[23] A. Roy, "Dynamic load balancing : improve efficiency in cloud computing," International Journal of Emerging Research in Management &Technology, vol. 9359, no. 4, pp. 78–82, 2013.

[24] A. A. P. Kazem, A. M. Rahmani, and H. H. Aghdam, "A modified simulated annealing algorithm for static task scheduling in grid computing," Int. Conf. Computer Science Information Technology, pp. 623–627, Aug. 2008.

[25] R. Lee and B. Jeng, "Load-balancing tactics in cloud," International Conference. Cyber- Enabled Distrib. Comput. Knowl. Discov., pp. 447–454, Oct. 2011.

[26] S. Lorpunmanee, M. N. Sap, A. H. Abdullah, and C. Chompoo-inwai, "An ant colony optimization for dynamic job scheduling in grid environment," proceedings of world academy of science, engineering and technology, volume 23 pp. 314–321, 2007.

[27] J. Kaur and S. Kinger, "A survey on load balancing techniques in cloud computing," International Journal of Science and Research (IJSR), vol. 3, no. 6, pp. 2662–2665, 2014.

[28] M. Zhou, R. Zhang, D. Zeng, and W. Qian, "Services in the cloud computing era : A Survey," Universal Communication Symposium (IUCS), 4th International, 2010.

[29] N. S. Raghava and D. Singh, "Comparative study on load balancing techniques in cloud computing," open journal of mobile computing and cloud computing, vol. 1, no. 1, 2014.

[30] B. Radojeviü and M. Žagar, "Analysis of issues with load balancing algorithms in hosted (cloud) environments." MIPRO, 2011Proceedings of the 34th International Convention, May 2011

[31] B. S. Rajeshwari, "Comprehensive study on load balancing," An international journal of advanced computer technology, vol. 3, no. Vi, pp. 900–907, 2014.

[32] M. Randles, D. Lamb, and a. Taleb-Bendiab, "A Comparative study into distributed load balancing algorithms for cloud computing," IEEE 24th Int. Conf. Adv. Inf. Netw. Appl. Work., pp. 551–556, 2010.

[33] D. Rastogi and F. U. Khan, "effective fault handling algorithm for load balancing using ant colony optimization in cloud computing," International Journal of Advances in Engineering & Technology, no. April, 2014.

[34] P. M. Shameem and R. S. Shaji, "A methodological survey on load balancing techniques in cloud Computing," International Journal of Engineering and Technology (IJET) vol. 5, no. 5, pp. 3801–3812, 2013.

[35] M. Abdullah and M. Othman, "Cost-based multi-qos job scheduling using divisible load theory in cloud computing," Procedia Computer Science, vol. 18, pp. 928–935, 013.

[36] C. T. Lin, "Comparative based analysis of scheduling algorithms for

resource management in cloud computing environment," International Journal of Computer Science International Journal of Computer Science and Engin and Engineering, " issue. 1,volume 1, pp. 17–23, 2013.

[37] R. P. Lingawar, M. V Srode, and M. M. Ghonge, "Survey on load-balancing techniques in cloud computing," International Journal of Advent Research in Computer & Electronics, vol. 1, no. 3, pp. 18–21, 2014.

[38] K. Nishant, P. Sharma, V. Krishna, C. Gupta, K. P. Singh, and R. Rastogi, "Load balancing of nodes in cloud using ant colony optimization," UKSim 14th Int. Conf. Comput. Model. Simul., pp. 3–8, Mar. 2012.

[39] J. Uma, V. Ramasamy, A. Kaleeswaran "Load Balancing Algorithms in Cloud Computing Environment - A Methodical Comparison " International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Volume 3, Issue 2, February 2014

[40] P. Samal and P. Mishra, "Analysis of variants in Round Robin Algorithms for load balancing in Cloud Computing,"(IJCSIT) International Journal of Computer Science and Information Technologies, vol. 4, no. 3, pp. 416–419, 2013.

[41] D. Powar, S. Swaroop Moharana, R. D. Ramesh "analysis of load balancers in cloud computing," International Journal of Computer Science and Engineering (IJCSE), vol. 2, no. 2, pp. 101–108, 2013.

[42] Yatendra Sahu, M. K. Pateriya "Cloud Computing Overview and load balancing algorithms", Internal Journal of Computer Application, Vol -65 No.24, 2013

[43] A. Agarwal and R. N. Milind, "Computing environment," Parallel, Distributed and Grid Computing (PDGC), 2014 International Conference, pp. 9–13, 2014.

[44] U. Bhoi, P. N. Ramanuj, and W. S. Email, "Enhanced max-min task scheduling algorithm in cloud computing," International Journal of Computer and Information Technology, vol. 2, no. 4, pp. 259–264, 2013.

[45] T. Desai and J. Prajapati, "A survey of various load balancing techniques and challenges in cloud computing," International Journal of Scientific & Technology Research, vol. 2, no. 11, 2013.

[46] X. Ren, R. Lin, and H. Zou, "A dynamic load balancing strategy for cloud computing platform based on exponential smoothing forecast."

[47] S. Ray and A. De Sarkar, "E xecution a nalysis o f l oad b alancing a lgorithms in cloud computing," International Journal on Cloud Computing: Services and Architecture (IJCCSA), vol. 2, no. 5, pp. 1–13, 2012.

[48] B. P. Rimal, E. Choi, and I. Lumb, "A taxonomy and survey of cloud computing systems," Fifth Int. Jt. Conf. INC, IMS IDC, pp. 44–51, 2009.

[49] V. Sakthivelmurugan, A. Saraswathi, and R. Shahana, "Enhancedload

balancing technique in public cloud," IJREAT International Journal of Research in Engineering & Advanced Technology vol. 2, no. 2, pp. 1–4, 2014.

[50] R. Somani and J. Ojha, "A hybrid approach for vm load balancing in cloud using cloudsim," International Journal of Science, Engineering and Technology Research (IJSETR), vol. 3, no. 6, pp. 1734–1739, 2014.

[51] R. Tong, "A load balancing strategy based on the combination of static and dynamic," Database Technology and Applications (DBTA), 2010 2nd International Workshop, no. 2, pp. 1–4, 2010.

[52] C.-M. Wu, R.-S. Chang, and H.-Y. Chan, "A green energy-efficient scheduling algorithm using the DVFS technique for cloud datacenters," Futur. Gener. Comput. Syst., vol. 37, pp. 141–147, Jul. 2014.

[53] Y. A. Younis, M. Merabti, and K. Kifayat, "Secure cloud computing for critical infrastructure : A Survey," 2013.

[54] M. Randles, D. Lamb, and 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, pp. 551–556. 2010

[55] M. Zhou, R. Zhang, W. Xie, W. Qian, and A. Zhou, "Security and privacy in cloud computing: a survey," in sixth int. conf. semant. knowl. grids, pp. 105–112, Nov. 2010.

[56] K. Al Nuaimi, N. Mohamed, M. Al Nuaimi, and J. Al-Jaroodi, "A survey of load balancing in cloud computing: challenges and algorithms," Second Symp. Netw. Cloud Comput. Appl., pp. 137–142, Dec. 2012.

[57] I. Ruiz-Agundez, Y. K. Penya, and P. G. Bringas, "A Flexible Accounting Model for Cloud Computing," in Annual SRII Global Conference, pp. 277– 284, 2011.

[58] Hung-Chang Hsiao, et al., "Load Rebalancing for Distributed File Systems in Clouds", IEEE transactions on parallel and distributed systems, VOL. 24, NO. 5, MAY 2015.

[59] D. B. L.D. and P. Venkata Krishna, "Honey bee behavior inspired load balancing of tasks in cloud computing environments," Appl. Soft Comput., vol. 13, no. 5, pp. 2292–2303, May 2013.

[60] H. Zhong, K. Tao, and X. Zhang, "An approach to optimized resource scheduling algorithm for open-source cloud systems," Fifth Annu. ChinaGrid Conf., pp. 124–129, Jul. 2010.

[61] R.-S. Chang, J.-S. Chang, and P.-S. Lin, "An ant algorithm for balanced job scheduling in grids," Futur. Gener. Comput. Syst., vol. 25, no. 1, pp. 20–27, Jan. 2009.

[62] 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.

[63] A. Daryapurkar and V. M. Deshmukh, "Efficient load balancing algorithm in cloud environment," International Journal Of Computer Science And Applications, vol. 6, no. 2, pp. 308–312, 2013.

[64] M. Kawser, "Performance comparison between round robin and proportional fair scheduling methods for LTE," Int. J. Inf. Electron. Eng., vol. 2, no. 5, 2012.

* * * * *