Article can be accessed online at http://www.publishingindia.com

Review of Optimize Load Balancing Algorithms in Cloud

Ankur Kushwaha*, Priya Pathak, Sandeep Gupta*****

Abstract

With cloud computing, new facilities in the information technology (IT) emerge from the convergence of occupational and technology viewpoints which furnish users entrance to IT resources anywhere and any time by pay-per-use fashion. Consequently, it should source eminent operative gain to the user and instantaneously ought to be beneficial for the cloud service provider. To achieve this goal, several challenges have to be confronted, where load balancing is single of them. The optimal choice of a resources for a specific job does not mean that the nominated resource persists enhanced for the entire execution of the job. The supply under loading/over-loading must be avoided which could be enlarged by appropriate load balancing mechanisms. But, to the best of our knowledge, in spite of the significance of load balancing methods and appliances, there is not any comprehensive and systematic review about and analyzing and studying its significant techniques. In this paper we study about cloud architecture or different load balancing technique at the end of our paper we compare four optimization based load balancing technique and gives the idea about new technique over existing ones.

Keywords: CC, VM, ACO, PSO, TLBPSO

Introduction

Cloud computing relating distributed technologies to content variability of applications and of user requirements. Share incomes, software, info through internet are the chief purposes of the cloud computing to abridged cost, well presentation and satisfy requirements. To recover the response time of the job, allocate the total load of the cooperative system. By this eliminating a condition during which a number of nodes are loaded whereas another are underneath full. Load balancing algorithms dose not taken the earlier state or conduct of the system, it depend upon the current conduct of the system since it is dynamic in the nature. Round robin algorithm process on circular order by handling the process without priority but equally spread current execution handle the process with priories. Throttled algorithm the client first needs the load balancer to discovery a appropriate VM to performed the essential operation. The architecture is completed creation for VM, fewer response time and min delay to transference. Therefore model calculable the virtual machine value and low information transfer value. This type of computational model promises to reduce the capital and operational cost of the client.

The entire execution time is valued in three stages. In the first stage the formation of the VM and they will be idle to come for the scheduler to schedule the jobs in the line, once jobs are payable, the VM in the cloud computing will start processing, which is the second stage, and lastly in the third stage the cleanup or the obliteration of the virtual machines. The quantity of the computing model can be valued as the entire number of jobs executed inside a time span lacking considering the VM destruction time and formation time[1]

* Department of computer Science, GICTS, RGPV, Gwalior, Madhya Pradesh, India. Email: Kushwaha.ankur@gmail.com
** Department of computer Science, GICTS, RGPV, Gwalior, Madhya Pradesh, India. Email: shakhi.priya@gmail.com
** Department of computer Science, Samrat Ashok Technical Institute, Vidisha, Madhya Pradesh, India. Email: Kushwaha.ankur@gmail.com

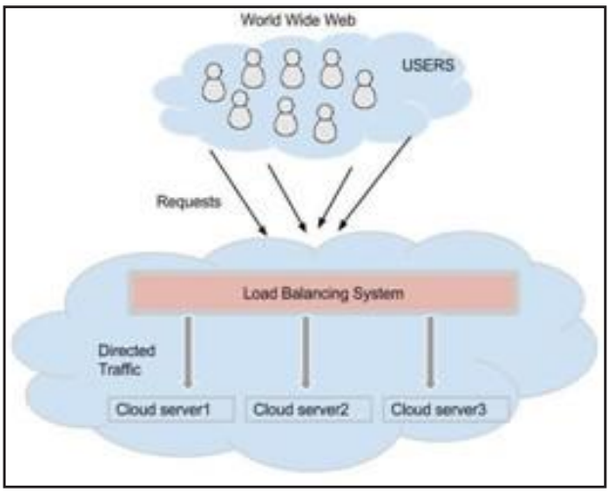


Fig. 1: Load Balancing in the Cloud Computing

Requirement of Load Balancing

Load balancing is a computer network technique for allocating assignments across multiple computing assets, for instance a computer cluster, processing units, computers, central, disk drives or network links,. Load balancing strategies to optimize supply use, max throughput, min response time, and avoid overload of any one of the assets. By the utilization of various mechanisms with load equalization rather than one element might growth dependableness through idleness. Load balancing in the cloud varies from traditional thinking on the load-balancing implementation and architecture by using product servers to achieve the load balancing as a result of it's troublesome to predict the quantity of requests that may be issue to a server. This provides for new opportunities and economies-of-scale, also presenting its own unique set of challenges. Load balancing is one of the central issues in cloud computing [8]. It is a mechanism that allocates the dynamic local job consistently crossways entirely the nodes in the entire cloud to evade a condition where some nodes are deeply loaded while others are idle or doing little work. It assistances to attained a high customer satisfaction and resources use ratio, thus improved the overall presentation and resource utilization of the system. It also ensure that each computing resources is dispersed professionally and fairly [9]. It further averts blockages of the system which may happen due to load inequity. When one or more components of any service stop working, load balancing facilitates in continuation of the service by implementing fair-over, i.e. de provisioning and in-provisioning of cases of applications without fail. depicts of the Load Balancing need in the cloud when there are requests from multiple customers. The present load balancing techniques in the clouds, consider many parameters such as response time, performance, throughput, scalability, fault tolerance, associated overhead, migration, utilization, and time resource The developing cloud computing model efforts to addressing the explosive development of handle massive amounts of data,web-connected devices and client demands. Thereby, giving rising to the question whether or not our cloud model is capable to balance the ever-increasing loaded in Associate in Nursing effectual approach or not.[2]

Load Balancing Algorithms

While pointing for improved load balancing and fulfillment of needs, the following aims need to be attained:

A. *Scalability:* The load balancing algorithm must provide scalability in terms of addition of new resources to address the everyday increasing demand of services with a greatly increasing number of users. This also demands flexibility in accommodating the change.

B. *Better response time:* The load balancing must be implemented and executed well enough to provide the best possible response to a user.

C. *Cost effectiveness:* A good load balancing algorithm must aim for better overall system performance with the cost being quite reasonable.

D. *Prioritization:* The tasks must be prioritized so that the critical tasks do not have to face the problem of starvation, or if addressed, the problem of late response.

E. *Fair node utilization:* The serving nodes must be utilized efficiently so that no single node is overwhelmed, leaving certain others totally free, or lightly loaded.

The load balancing algorithms that area unit presently being utilized in cloud computing is delineated below, alongside sure considerations:

A. *Random:* The random load balancing algorithm is static (4) in nature, it being generally defined in the design or implementation of the system. It selects the node randomly by making use of a random number generator (5).

B. *FCFS:* The First Come First Serve algorithm (6) is a simple load balancing technique wherein each load balancer conserves a job line in which job stays for its turn to get implementation. The advantages of FCFS are

a result of its being fast and simple. But it results in a poorer overall response time in case smaller tasks have to wait for longer time because of being at a later place in the queue.

C. *Round Robin:* The Round Robin algorithm (4) allocates the nodes to fulfill the requests in a round robin manner for a definite time slice, i.e. according to the method distribution order that is preserved locally. This serves the benefit of fast response in the case of equal workload distribution among the methods. However, the job processing time for dissimilar methods is not the similar. So, certain nodes may be deeply loaded but some others may continue idle.

D. *Weighted RR:* In this method, a ration weight is defined for each machine. According to this algorithm (5), the number of networks that every machine accepts over time is balanced to the define ratio weight. This algorithm improved the Round Robin because whenever we define weighted assignments like "Machine 1 is able to serve 3x the load that machines 2 and 3 are able to handle"; three requests are sent by the load balancer to machine 1 for each request to the others. This algorithm works smoothly but has a problem because of the static definition of the weights in the beginning.

E. *Dynamic Round Robin:* This algorithm (5) is quite same to Weighted Round Robin; the change being that the servers are unceasingly monitored and the weights keep on altering. This is a dynamic load balancing technique. It makes use of many features of the real-time server presentation analysis, like the present number of the connections each node or the fastest node response time to allocate the connections. The only matter with this algorithm is that it is infrequently accessible in a simple load balancer, because of its dynamic nature.

F. *Equally Spread Current Execution Load:* This algorithm (7) requires continuous monitoring of jobs which are present for execution in order to queue up the jobs and pointer them ended to diverse virtual machines. The load is dispersed randomly by inspection the size and thereby moving the load to that VM which is casually loaded or can handled that task simply and takes the less time, while giving max amount.

G. *Throttled Load Balancing Algorithm:* In this algorithm (8), when a client request is receiving, the load balancer stabs to discover a suitable VM to performed the required operation. The algorithmic method starts by maintaining a listing of the whole accessible VMs. assortment is performed so as to hurry up the operation method. The request from a client is accepted if a match is found on the basis of size and availability of the machine. The VM is then allocated to the client. If, however VM that matches the criteria is available, then the load balancer queues up the request.

H. *Min-Min Algorithm:* This algorithm (4) begins by discovery the least completion period for wholly tasks. Then among these min times, the min value among entirely the tasks on any supply is chooses and accordingly the tasks is scheduled on the equivalent machine. Thereafter, the execution time of the assigned task is added to the execution times of other tasks on that machine to update the execution time on that machine. The allocated task is then detached from the list of tasks that are yet to be allocated to the machineries. This procedure is followed till all the tasks area unit appointed on the resources. the foremost downside of this technique is that it will cause starvation (9).

I. *Max-Min Algorithm:* Max-Min algorithm (9) is same to the min-min algorithm except that after discovery out min execution times, the max value is designated. This is the maximum time among stentirely the tasks on any resources, according to which the task is arranged on the equivalent machine. Thereafter, the execution time of the assigned task is added to the execution times of other tasks on that machine to update the execution time on that machine. The allocated task is then detached from the list of responsibilities that are yet to be allocated to the machineries. This procedure is followed until all the tasks are assigned on the resources.

J. *Token Routing:* The algorithm (4) was designed with an aim of minimizing the system cost by moving tokens around the system. Due to communiqué bottleneck, agents cannot own enough info of allocating workload. The drawback of this algorithm can be removed with the help of heuristic approach of token based load balancing, thereby providing fast and efficient routing decisions. Here, managers do not essential to have completed knowledge of their global state and neighbors' working load, then make their owned choices on where to passed the token by actually building their own knowledge based. In this approach no communication overhead is generated because the content is really derived from the antecedently received tokens.[3]

***j. Genetic Algorithm*:** The load balancing abides by three rules such as the location rule, the distribution rule, and the selection rule. Here the work will method through a dynamic method after doing scheduling server. Initially the tasks will be secure a number. Afterward it will auto

executed task size arbitrarily. Then the task handled from a task slot, where the randomly generator are deposited for processed. Mainly the development of cloud computing is dynamic but for the arrangement purpose it can be expressed as allocating N number of jobs applied by the cloud customers to M number of processing unit in cloud. Every processing unit has a processing unit vector (PUV), where vector consists of MIPMS that mean how many million instructions can be processed by the machine per second. R is indicated as delay cost and x is cost of execution of instruction

K. Ant Colony Optimization: Separate ants are behaviorally much unworldly bugs. They have a very limited memory and exhibit individual behavior that appears to have a large random component. Acting as a shared though, ants achieve to perform a change of complicated tasks with great consistency and steadiness. While this is basically self-organization before learning, ants have to manage with a marvel that looks desperately like overtraining in strengthening learning techniques algorithmic method founded on ant conduct. The attempt to develop algorithms inspired by one aspect of ant behavior, the ability to find what computer scientists would call shortest paths, has become the field of the ACO, the most creative and broadly recognized

L. Particle Swarm Optimization: Task founded System Load Balancing method through PSO (TBSLBPSO) to attained scheme load balancing by individual moving additional tasks from an loaded VM instead of migrating the whole overloaded VM. PSO optimization model was used to migrate the additional tasks to the novel host VMs. It exposed that the TBSLB-PSO procedure expressively reductions the time taken for the load balancing technique associated to traditional load balancing procedures and the loaded VMs was not be stopped during the migration technique, and there was no vital to use the VM pre-copy technique. It eliminated VM downtime and the risk of losing the last activity performed by a customer, and increased the Quality of Service experienced by cloud customers.

M. Firefly Algorithm: Currently we can idealize some of the irregular features of the fireflies so as to produce firefly enthused algorithms. For simplicity in defining our novel Firefly Algorithm (FA), we now use the succeeding three idealized rules: (1) entirely fireflies are unisex so as to one firefly will be complicated to other fireflies regardless of their sex; (2) Attractiveness is relative to their light, thus for any two irregular fireflies, the less happier one will transmission to the brighter one. The attractiveness is relative to the illumination and they both decrease as their distance developments. If there is no brighter one than a specific firefly, it will moved arbitrarily; (3) The illumination of a firefly is pretentious or determined by the landscape of the impartial function.

Based on metrics discussed in section 3, the present load balancing methods Comparison of presenting load balancing techniques [4]

Metrics	*Honeybee Scheduling*	*Biased random Sampling*	*Active clustering*	*OLB+ LBMM*	*Join Idle Queue*	*Min- min*	*Min-max*
Throughput	No	No	No	No	No	Yes	Yes
Overhead	No	Yes	Yes	No	Yes	Yes	Yes
Fault tolerance	No	No	No	No	No	No	No
Migration Time	No	No	Yes	No	No	No	No
Response Time	No	No	No	No	Yes	Yes	Yes
Resource utilization	Yes	Yes	Yes	Yes	No	Yes	Yes
Scalability	No	No	No	No	No	No	No
Performance	No	Yes	No	Yes	Yes	Yes	Yes

Challenges of Load Balancing

Overhead Associated- It is defines the amount of above involved though applying a load-balancing system. It is collected of the overhead due to drive of tasks, interposes communication. Overhead should be abridged so that a load balancing algorithm achieves good.

Throughput - It is the number of task executed in the fixed interval of time. To improve the performance of the system, throughput should be high.

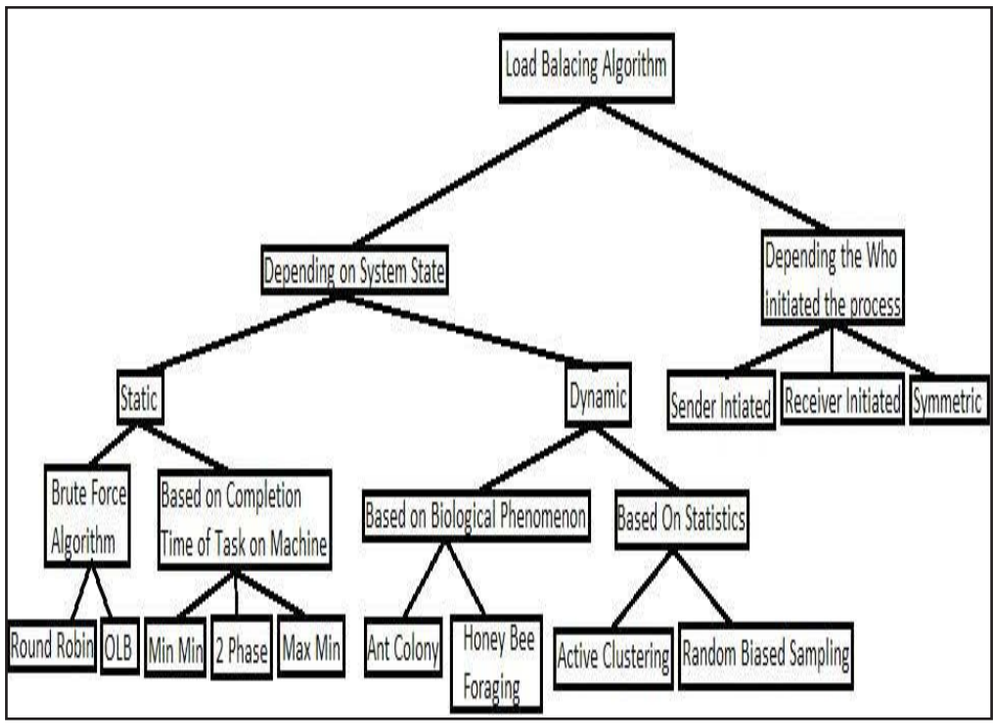
Performance - It can be defined as the efficiency of the system. It must be improved

Resource Utilization - It is used test the utilization of resources. It should be max for an efficient load balancing system.

Scalability - In the quality of service should be same if the number of users increases. The more number of nodes can be added without affecting the service.

Response Time - It is defined as the amount of your time taken to respond by a load balancing algorithmic rule in a very distributed system. For well presentation, this parameter should be shortened. **Fault Tolerance** –In spite of the node failure, the capability of an system to performed uniformed load balancing. The load balancing is the best fault-tolerant technique.

Point of Failure - Designed the system in such a way that the single point failure does not affect the provisioning of services. Like in federal system, if single central node is failed, then the entire system would fail, so load balancing system must be intended so as to overcome this difficulties.



Based on this state of the system, load balancing algorithms is classified into 2 types:

Static Algorithm: This standing of the node isn't taken into thought [3]. All the nodes and their properties area unit illustrious earlier. It is supported this previous information, the algorithmic rule works. Since it does not use current system status information, it is easy to implement.

Dynamic Algorithm: This kind of algorithm relies on this standing of the system [3]. The algorithm workings according to the dynamic modifications in the state of nodes. Status Table keeps the Present status of entirely nodes in the cloud. Dynamic algorithms are compound to implementation but it balanced the load in real manner.

Based on the creator of the algorithm, Load Balancing algorithms can be classified into three types [2]:

Receiver Originated: The disorder of the Load balancing condition can be recognized by the receiver/server in the cloud and that the server pledges the execution of the Load Balancing algorithm. **Sender Originated**: Sender classifies that the nodes are in huge number so that the sender recruits the execution of the Load Balancing algorithm.

Symmetric: It is the grouping of both the receiver initiated and sender initiated types. [5]

Related Works

Rajesh et al. In this paper, reviews from various authors have given in regards with cloud computing for good load balancing. Various techniques as well as methodologies are given their with their studies and results from their experiments. Decent load balance will recover the presentation of the whole cloud. Though, there is no common technique that can familiarize to all possible dissimilar situations. Various methods have been developed in improving existing solutions to resolve new problems. Every particular technique has benefit in a specific area but not in entirely circumstances. Therefore, the present study integrates numerous approaches and switches among the load balance techniques based on the system position. A relatively simple method can be used for the partition idle state with a more complex method for the normal state. The load balancers then switch methods\ as the standing changes.[6]

Sethi et al. Cloud computing is an increasing area in the research and industry today, which includes distributed computing, virtualization, software internet, and web services. A cloud contains of numerous elements such as data centers clients, and dispersed servers, internet and it includes high availability, fault tolerance, scalability, flexibility, abridged above for users, on demand services abridged cost of ownership, and etc. The facilities of the cloud computing are gratifying ubiquitous, and serve as the main source of the computing power for different applications like private computing applications and enterprises. In this paper we familiarized the novel load balancing algorithm by means of fuzzy logic in the cloud computing, in which load balancing is a core and challenging and problems in the Cloud Computing. The processor rapidity and allocated load of Virtual Machine (VM) are using to balance the load in cloud computing over fuzzy logic.[7]

R. Barani et al. We have replicated two dissimilar dynamic load balancing algorithms for implementing the user request in the cloud environment. Every algorithm is experiential and their scheduling standards like data center service time, average response time and total cost of diverse data centers are create. Here we have used 6 User Bases and 3 Data centers and associated the presentation of algorithms by 25, 50 and 75VMs and outcome is as shown in graph. Our future work is to grow an adaptive algorithm appropriate for varied environment such that it handled Big Data and recover complete response time with abridged price.[8]

Dimri et al. In the cloud computing environment, load balancing is one of the major issues that is highly needed to distribute local workload to all the nodes in the cloud to recover the performance and max resource utilization. This paper described cloud computing, types of load balancing algorithms, load balancing metrics, load balancing, and appliances of dynamic load balancing algorithms. This paper primarily focused on the dynamic load balancing algorithm in the cloud environment. For this, many prevailing dynamic load balancing algorithms are survey. By comparing the algorithms on different metrics we tried to find the scope for improving throughput, fault tolerance, performance, response time, migration time, decreasing supply utilization and above in the load balancing algorithm. Future work is connected to scheming a novel dynamic load balancing algorithm with fault tolerance for well minimum response time, resource utilization and fast throughput of the cloud computing environment.[9]

Mayur et al. In this paper we have discussed various soft computing techniques those are used for load balancing in cloud computing and are still to be improved and unexplored. In this paper we have also deliberated the evaluation of several algorithms founded on their execution environment. One of the main issue of the cloud computing is load balancing because overloading of the system may lead towards the poor performance of the system. So such more efficient work is still to be done for load balancing in cloud. Our paper focuses on diverse load balancing algorithms which used soft computing method in the cloud computing environment. Further this can be simulated with cloud sim and the graphical representation and results can be explored. Some of them are modified or enhanced versions of previously defined or discovered algorithms. Some of algorithms are still lacking of improvements are to be done on them. So here we are discussing the comparative study of them and various environments are defined and compared with each other. [10]

Simulation and Result

Simulation of these optimize algorithm we done on cloudsim we simulate PSO, TLBPSO, ACO, Firefly

Time of execution: below table shows that what time process take to execute on virtual machines on data center below table show time of execution with different algorithms on varying the VMs and cloudlets.

No of VMs	*No of cloudlet*	*PSO*	*ACO*	*Firefly*	*Time taken TLBPSO*
4	8	45	42	43	40.02
6	12	118	110.90	109.89	102.47
8	16	78	70.98`	69.90	64.15
10	20	100	98.90	99.0	97.66

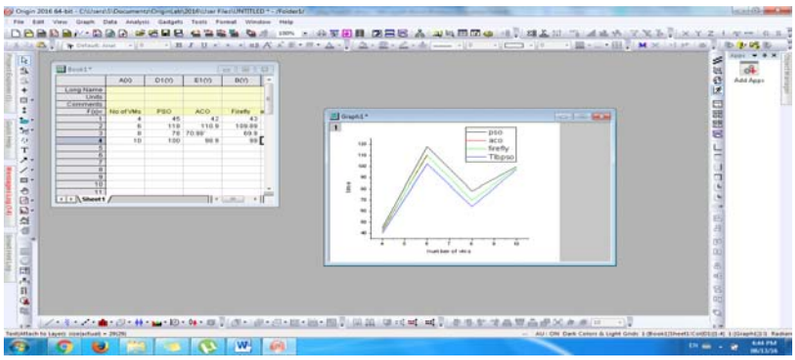


Fig. 2: Time of Execution in Different VMs

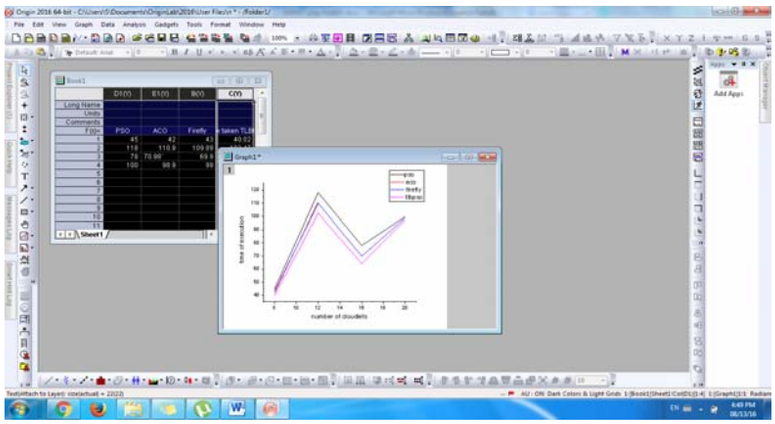


Fig. 3: Time of Execution with Different Cloudlets

By seeing above graphs we can say that TLBPSO perform better compare to all that mean time taken by TLBPSO is less compare to all

CPU utilization rate: number of cloudlets execute on VMs if VM execute more task than CPU utilization is high and performance is increase. Below table show CPU utilization with different optimization algorithm.

No of VMs	*PSO*	*ACO*	*Firefly*	*Time taken TLBPSO*
70	1.0	1.8	4	8
80	1.5	2.5	5	10
90	2	3	6	15
100	3	4	7	20

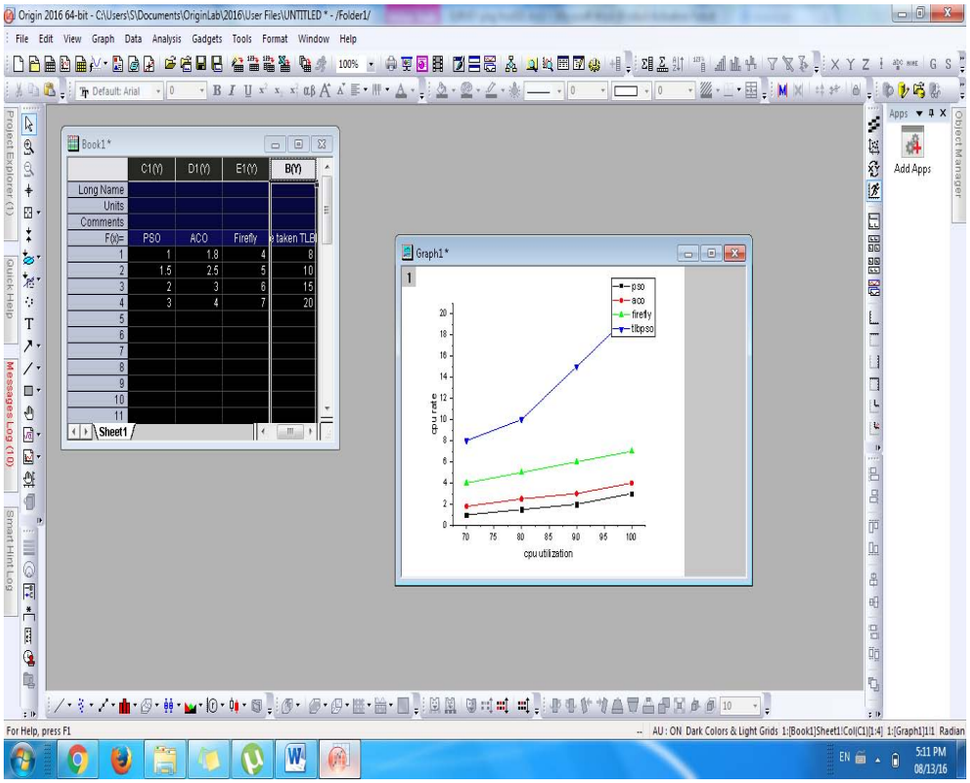


Fig. 4: CPU Utilization

By seeing above graph we can say that cpu utilization is better in TLBPSO performance of TLBPSO is good compare to all.

Conclusion

This study presented a methodical evaluation of the load balancing methods in the cloud environments. In a related way, we studied various state-of-the-art load balancing in the cloud computing scheme. Through responses provoked by three investigative research queries, we found proof checking load balancing as an ascending device that presents a novel paradigm by increasing cloud performance and impacting on resource utilization, also, ensures that each input request is distributed efficiently and fairly. Based on the accessible literature, we decided to categorize the field into two subdomains connected to the hybrid load balancing and dynamic load balancing studies. We also discussed the disadvantages and advantages related with numerous load balancing algorithms. The trials of these algorithms are addressed so that more effectual load balancing methods can be advanced in future. Proper load balancing have the ability to keep minimum resource consumption which will conclude further reduction of energy consumption and carbon emission rate. In general, load balancing devices in the computing environment still essential developments in terms of managing the heterogeneity of its environment so as to become a truly on-demand method, reducing associated overhead, facility response time and improving presentation etc. The complete data composed in this study help to inform the researchers with the state-of-the-art in the load balancing field. exclusively, the answers to the defined questions summarized load balancing's primary purpose, current challenges, open issues, approaches and mechanisms in cloud systems. We honestly hope that the results of this study will help investigators to grow novel research façades that additional donate to the maturity and acceptance of load balancing in the cloud computing.

References

Nitika., Shaveta., & Raj, G. (2012). Comparative analysis of load balancing algorithms in cloud computing. *International Journal of Advanced Research in Computer Engineering & Technology*, May, 1(3), 120-124.

Haryani, N., & Jagli, D. (2014). Dynamic method for load balancing in cloud computing. *IOSR Journal of Computer Engineering*, July-August, 16(4), 23-28.

Brar, H. R., Thapar, V., & Kishor, K. (2014). A survey of load balancing algorithms in cloud computing. *International Journal of Computer Science Trends and Technology*, May-June, 2(3), 103-106.

Sran, N., & Kaur, K. (2013). Comparative analysis of existing load balancing techniques in cloud computing. *International Journal of Engineering Science Invention*, January, 2(1), 60-63.

Kashyap, D., & Viradiya, J. (2014). A survey of various load balancing algorithms in cloud computing. International Journal of Scientific & Technology Research, November, 3(11), 115-119.

Kumar, R., & Singh, C. (2015). Survey: Cloud partitioning using load balancing approach for public cloud infrastructure. International Journal of Engineering Sciences & Research Technology, April, 4(4), 170-176.

Sethi, S., Sahu, A., & Jena, S. K. (2012). Efficient load balancing in cloud computing using fuzzy logic. *IOSR Journal of Engineering*, July, 2(7), 65-71.

Suguna, S., & Barani, R. (2015). Simulation of dynamic load balancing algorithma bonfring. *International Journal of Software Engineering and Soft Computing*, July, 5(1), 1-6.

Dimri, S. C. (2015). Various load balancing algorithms in cloud environment. *International Journal of Emerging Research in Management &Technology,* July, 4(7), 263-266.

Pilavare, M. S., & Desai, A. (2015). A survey of soft computing techniques based load balancing in cloud computing. *International Journal of Computer Applications*, January, 110(14), 1252-1256.