IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

Load Balancing Techniques In Cloud Computing: Review Paper

[1]Foram Patel, [2]Dr. Mohit Bhadla

[1]Student, [2] HOD

[1]Computer Department,

[1]Gandhinagar Univrsity, Ahmedabad, India

Abstract: Cloud computing is a model used to provide like IT services and resources on demand through Internet. The model offers platform for deployment, convenient access to web services, storage and so on to user or organization to purchase the required services as per their requirements. Load Balancing is a key aspect in the field of cloud computing which works as load distributer among available nodes for processing and avoid the situation in which some node becomes overloaded while other nodes in network are free or have little work to complete. It is a common issue in cloud computing which affects the performance of the application, Quality of Service (QOS) measures and Service Level Agreement (SLA) document of cloud providers. This Paper presents a literature survey of various existing load balancing methodologies or algorithms in different forms such as static, dynamic, nature inspired in cloud environment which is mainly focused on response time, virtual machine costing, fault tolerance and throughput. This survey will be beneficial for future researchers to review and compare different types of load balancing algorithm and find the insight to identify research gaps as inspiration for new innovations in the field of load balancing in cloud computing.

Index Terms - Load Balancing, Virtualization, Virtual Machine, Load Balancing Algorithm, Coud Computing.

I. INTRODUCTION

Rapid development in technology replaced traditional computing methods with cloud computing. It allows consumers to connect many configurable assets like processor, memory, networks, application etc. to provide facilities to consumers as pay-per-use rates [8]. In cloud model user access the service based on user requirements regardless of cloud or user location. In fact, instead of installing their applications on their PCs, users can receive them from the internet in the form of applications and installed on a set of network servers. In other words, only by access to a mobile device, computer, etc., they can be connected quickly and get access to the cloud services. The major cloud companies such as Google, Amazon and Microsoft.

Innovation for cloud computing is critically facilitated by virtualization. Hardware and software approaches that allow physical systems to be partitioned among numerous virtual instances that function simultaneously and share the base physical or bare metal resources and equipment draw attention. The broad scope of research into all key domains and enterprise applications is enabled by the fusion of Cloud computing and Virtualization technologies, as observed in various references.

In cloud computing field, Load Balancing refers to optimizing and efficient distribution of traffic load among the available nodes using concept of virtualization of resources. It ensures equitable and dynamic task allocation while effectively utilizing resources for executing end user requests.

Load balancing is the efficient and fair assignment of work to computing resources to maintain the load satisfaction (i.e., processor load, the used memory, delays, or the network load) of users and increase the rate of resource productivity [1]. When any virtual machines (VMs) in the network have overhead volumes than others, the system's efficiency is reduced. Thus, the cloud would be constraining, and the time of completing

different tasks would be afflicted. In addition, when the cloud system is faced with a multiple significant number of demands, the system has to assign them among its resources. If the cloud system can provide users with productive, convenient and accessible resources to complete such tasks, which will strengthen the system performance [9]. In a cloud system, various computing resources exist to facilitate responding to users' demands. Thus, the proper selection of resources by considering the characteristics of tasks will enhance a system's efficiency; therefore, a mechanism is required to select appropriate resources in responding to user demands [1].

II. LITERATURE REVIEW

Sefati s [1] proposed the grey wolf optimization algorithm which has been used based on resource reliability capacity to maintain proper load balancing among the nodes in the network. This paper grey wolf optimization (GWO) process method has been shown in which first the GWO algorithm tries to find the unemployed or any busy nodes in the network, after discovery it tries to calculate threshold and fitness function of each node. The results proves that cost and response time in proposed method are less than pre-existing methods and obtain the positive solution out of it.

Sahid, M.A [2] presents the performance evaluation of exiting load balancing algorithm which are particle swarm optimization (PSO), Round Robin, Equally Spread Current Execution (ESCE) and throttled load balancing, as well as service broker policies which were CDC, optimize response time, and reconfigure dynamically with load. This paper offers a detailed performance by employing a cloud analyst platform.

Safiq, D.A [3] presents a review study of various load balancing techniques in a static dynamic and nature inspired cloud environment. The problem related load balancing was discussed through comparative analysis of proposed algorithms by researchers, by which author categorized the literature based on research gaps. The comparative analysis is represented in tabular format containing pros, cons, simulation tools, publication year, author name. which can become base for the future researchers for improvements and innovations in the field of load balancing.

Ghomi, E.J [4] presents the study on task scheduling and load balancing algorithms and presents a new classification of such algorithms like Hadoop MapReduce load balancing category, Natural Phenomena-based load balancing category, Agent-based load balancing category, General load balancing category, application-oriented category, network-aware category, and workflow specific category. In each category, we studied some techniques and analysed them in terms of some metrics and summarized the results in tables. Key ideas, primary goal, merits, demerits, assessment techniques, publication year were metrics that we considered for load balancing techniques. Recently, load balancing techniques are concentrating on major two critical metrics, that is, energy saving and reducing carbon cost. Future work suggested by author are first study and analyze more recent techniques in each of proposed categories, second evaluate each technique in a simulation toolkit and compare them based on new metrics.

Ijeoma, C. [9] presents a detail review on hybrid load balancing algorithms. Many static and dynamic load balancing algorithm have been proposed and implemented in the past but have not provide required and efficient results comparatively. So, hybrid algorithm comes to rescue for overcoming the limitation of both static and dynamic algorithms individually by inheriting the properties of both and creating combination out of it which provide effective and optimized solutions. It provides proper resource utilization and reduces throughput time of a task which also leads cost cutting and meeting customer requirements and satisfaction

III. LOAD BALANCING MODEL

In the environment of cloud computing, a large scale of users and data centers has been distributed all over the world. When multiple user requests and made simultaneously to the cloud for the service, the cloud has to handle every user request efficiently and provide satisfactory results to the user. The cloud must distribute the load among available resources fairly which require working load to be distributed across all nodes by using appropriate load balancing algorithms.

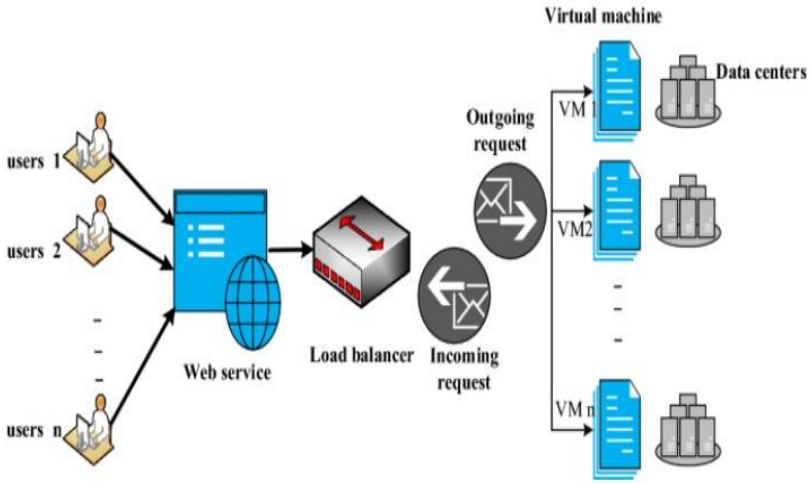

fig 1 load balancing model [6]

A client sends request to the internet which is received by load balancer which uses load balancing algorithms for selecting the most suitable server from the available pool based on factors like server load, health status, and capacity. The load balancer forwards request to selected server which processes the request and generates a response. The server sends the response back to load balancer to forward it to the client.

Load balancing using Virtualization

Virtualization is a dominant technology in cloud computing. The main objective of virtualization is sharing expensive hardware among VMs. VM is a software implementation of a computer that operating systems and applications can run on. VMs process the requests of the users. Users are located all around the world and their requests are submitted randomly. Requests have to be assigned to VMs for processing. Therefore, the task assignment is a significant issue in cloud computing. If some VMs are overloaded while others are idle or have a little work to do, QoS will decrease. With the decreasing of QoS, users become unsatisfied and may leave the system and never return. A hypervisor or Virtual Machine Monitor (VMM) is used to create and manage the VMs.

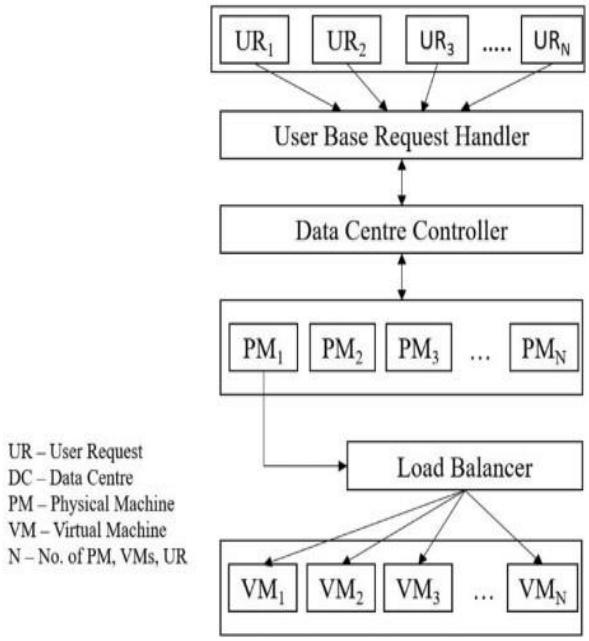

Fig 2 Load Balancing Using Virtualization [8]

[1] Here User Request (UR) contains different task (T) which should be executed by the server. The UR function for it can be:

$$UR_n = \{T_1, T_2, \ldots, Tn\}$$

In Addition, any Physical Machine (PM) is collection of N no. of Virtual Machine (VM):

$$PM_n = \{VM_1, VM_2, \ldots, VMn\}$$

Cloud system is consisting of Physical Machines (PM) for task execution by providing independent user experience on single machine to each user using virtualization:

$$C = \{PM_1, PM_2, \ldots, PM_n\}$$

In load balancing using virtualization technique model we can see the load balancer receives user requests and run load balancing algorithms to distribute the user request among the available Virtual Machines (VMs). The load balancer decides which VM should be assigned to the next request. The data centre controller is in charge of task management. Tasks are submitted to the load balancer, which performs load-balancing algorithm to assign tasks to a suitable VM. VM manager is in charge of all VMs available in the network.

Load balancing metrics

1. Throughout – This metrics is used to indicate number of jobs and request appropriately completed in VM in unit time interval [7].
2. Response Time – It measure total time that system required to serve a submitted task in reference of generated query [4].
3. Request Queue Length – The length of the queue on each server helps determine load of a particular server [6].
4. Makespan – It is used to calculate maximum completion time when the resources are allocated to a user [4].
5. Migration Time – Total time required to transfer a task from one overloaded node to other under loaded node [4].
6. Scalability – Uncertain variation in the number of user request and computational load which affects the system performance should be handle using efficient load balancing algorithm [7].
7. Fault Tolerance – Determines the capability of the algorithm to handle load balancing in the cause of some failure in particular node in the network [4].
8. Resource Utilization – This is analysed to guarantee the correct use of all resources in the system [8].

Challenges in load balancing

1. Virtual machine migration – The service on demand nature of cloud computing implies that when there is a service request, the resources should be provided. Sometimes resources should be migrated from one physical server to another, possible on a far location [4].
2. Spactically distribution nodes – Nodes in cloud computing are distributed geographically. The challenge in this case is that the load balancing algorithms should be designed so that they consider parameters such as the network bandwidth, communication speeds, the distances among nodes, and the distance between the client and resources [4].
3. Single point failure - Some of the load-balancing algorithms are centralized. In such cases, if the node executing the algorithm (controller) fails, the whole system will crash because of that single point of failure. The challenge here is to design distributed or decentralized algorithms [4,1].
4. Algorithm Complexity - The load-balancing algorithms should be simple in terms of implementation and operation. Complex algorithms have negative effects on the whole performance [4].
5. Emergence of small data centres - small data centres are cheaper and consume less energy with respect to large data centres. Therefore, computing resources are distributed all around the world. The challenge here is to design load-balancing algorithms for an adequate response time [4].

Classification of Load Balancing Algorithms

Load balancing Methodologies are mainly classified based on resource initiation and deployment. LB are mainly classifying into 3 types which are:

1. *Static Algorithm*

Static LB Algorithm do not consider the system's condition or metrics, such as processing while distributing requests. These algorithms distribute requests evenly across VMs or according to other principles not influenced by constraints, thus ideal for systems with hardly any change in load [7]. Static load balance strategies need not know the system present state only machine resources such as run time, space, storage capabilities and node handling capability are kept in progress [8].

Table 1 Static Algorithms [2,3,4,5,8]

Algorithm Name	Description	Advantages	Disadvantages
Round Robin	Distributes requests cyclically among servers.	Simple to implement, fair distribution.	May not be optimal for servers with varying capacities.
Random	Assigns requests randomly to servers.	Simple to implement.	May not be efficient for servers with varying capacities.

2. *Dynamic Algorithm*

The cloud provider deploys applications in a dynamic environment. For dynamic LB algorithms, various research challenges must be considered, such as how often resource scheduling must be called, which node leads the LB, obtaining VM load information, and load migration across nodes [7].

Table 2 Dynamic Algorithms [2,3,4,5,8]

Algorithm Name	Description	Advantages	Disadvantages
Least Connection	Directs requests to the server with the less active connections.	Efficient for servers with varying capacities.	May not be optimal for servers with different processing speeds.
Least Response Time	Routes requests to the server with the shortest average response time.	Optimizes performance for latency-sensitive applications.	May not be efficient for servers with different processing speeds.
Weighted Round Robin	Assigns weights to servers and distributes requests proportionally to their weights.	Allows for customization based on server capacities.	Requires knowledge of server capacities.
Weighted Least Connection	Combines least connections and weighted round robin.	Efficient for servers with varying capacities and processing speeds.	Requires knowledge of server capacities and processing speeds.

3. Nature Inspired Algorithm

Nature-inspired load balancing algorithms draw inspiration from natural phenomena and biological systems to design effective load distribution strategies. These algorithms often exhibit robustness, adaptability, and the ability to handle complex and dynamic environments.

- **Ant Colony Optimization (ACO):** Inspired by the behavior of ants foraging for food, ACO uses pheromones to guide agents (ants) towards optimal paths. In load balancing, pheromones can represent the load on servers, with agents preferring paths leading to less loaded servers.
- **Particle Swarm Optimization (PSO)**: Inspired by bird flocking behavior, PSO uses a population of agents that move through the search space. Particles are affected by their own observation and the experiences of others to find optimal solutions. In load balancing, particles can represent different load distribution strategies, and their fitness can be evaluated based on metrics like response time and resource utilization.
- **Genetic Algorithm (GA):** Inspired by natural selection, GA uses a population of solutions (chromosomes) that evolve through processes of selection, crossover, and mutation. In load balancing, chromosomes can represent different load distribution strategies, and the fittest ones are selected to reproduce and create new offspring.
- **Firefly Algorithm (FA):** Inspired by the behavior of fireflies, FA uses a population of agents (fireflies) that move towards brighter fireflies, representing better solutions. In load balancing, fireflies can represent different servers, and their brightness can be determined by their load or performance metrics.
- **Harmony Search (HS):** Inspired by music improvisation, HS uses a population of solutions (harmonies) that are created and improved through improvisation and memory. In load balancing, harmonies can represent different load distribution strategies, and they are evaluated based on their fitness.
- **Bacterial Foraging Optimization (BFO):** Inspired by the foraging behavior of bacteria, BFO uses a population of agents (bacteria) that move and reproduce based on nutrient concentration. In load balancing, bacteria can represent different servers, and nutrient concentration can represent their load or performance metrics.
- **Cuckoo Search (CS):** Inspired by the behavior of cuckoos laying eggs in other birds' nests, CS uses a population of agents (cuckoos) that lay eggs (solutions) in other birds' nests (solutions). The best solutions survive. In load balancing, cuckoos can represent different load distribution strategies.
- **Artificial Bee Colony (ABC):** Inspired by the behavior of honey bees foraging for food, ABC uses a population of agents (bees) that search for food (solutions) and share information with others. In load balancing, bees can represent different servers, and food can represent their load performance

IV. CONCLUSION

Load Balancing distributes workload effectively across all the nodes in the cloud to achieve high resource utilization and user satisfaction by avoiding situation where some nodes are either highly overloaded or idle. Therefore, overall implementation and resources utilization is managed gracefully. A large number of Methodologies and algorithms have been proposed to fix issues related to load balancing such as: Tasks scheduling, migration approach, resource utilization etc. This study provides analysis of various load balancing methodologies with strategies indicating its principle, pros and cons. This taxonomy of load balancing algorithm will help to find depth analysis of different load balancing algorithms in cloud computing. This paper provides survey work for improvising load balancing using virtualization in cloud computing field. The sole goal of proposed approach is to provide availability of processing node to user for executing task. It avoids system failure due to overloading and optimizing storage issue in cloud using virtualization and migration technique. It creates virtual instances of system for creating replica of a system for the end user which can migrate task to one node to another due to any inconvenience without letting it know to the client.

REFERENCES

[1] Sefati, S., Mousavinasab, M., & Zareh Farkhady, R. (2022). Load balancing in cloud computing environment using the Grey wolf optimization algorithm based on the reliability: performance evaluation. *The Journal of Supercomputing*, *78*(1), 18-42.

[2] Shahid, M. A., Alam, M. M., & Su'ud, M. M. (2023). Performance evaluation of load-balancing algorithms with different service broker policies for cloud computing. *Applied Sciences*, *13*(3), 1586.

[3] Shafiq, D. A., Jhanjhi, N. Z., & Abdullah, A. (2022). Load balancing techniques in cloud computing environment: A review. *Journal of King Saud University-Computer and Information Sciences*, *34*(7), 3910-3933.

[4] Ghomi, E. J., Rahmani, A. M., & Qader, N. N. (2017). Load-balancing algorithms in cloud computing: A survey. *Journal of Network and Computer Applications*, *88*, 50-71.

[5] Afzal, S., & Kavitha, G. (2019). Load balancing in cloud computing–A hierarchical taxonomical classification. *Journal of Cloud Computing*, *8*(1), 1-24.

[6] Laha, J., Pattnaik, S., & Chaudhury, K. S. (2024). Dynamic Load Balancing in Cloud Computing: A Review and a Novel Approach. *EAI Endorsed Transactions on Internet of Things*, *10*.

[7] Lohumi, Y., Gangodkar, D., Srivastava, P., Khan, M. Z., Alahmadi, A., & Alahmadi, A. H. (2023). Load Balancing in Cloud Environment: A State-of-the-Art Review. *IEEE Access*, *11*, 134517-134530.

[8] Kulkarni, M., Deshpande, P., Nalbalwar, S., & Nandgaonkar, A. (2022). Taxonomy of Load Balancing Practices in the Cloud Computing Paradigm. *International Journal of Information Retrieval Research (IJIRR)*, *12*(3),300292

[9] Ijeoma, C. C., Samuel, A., Okechukwu, O. M., & Chinedu, A. D. (2022). Review of hybrid load balancing algorithms in cloud computing environment. *arXiv preprint arXiv:2202.13181*.

[10] Sasidhar, T., Havisha, V., Koushik, S., Deep, M., & Reddy, V. (2016). Load Balancing Techniques for Efficient Traffic Management in Cloud Environment. *International Journal of Electrical and Computer Engineering (IJECE)*, *6*(3), 963-973.