IRAQI Academic Scientific Journals
Samarra Journal of Pure and Applied Science
SJPAS
مجلة سامراء للعلوم الصرفة والتطبيقية
www.sjpas.com
p ISSN: 2663-7405
e ISSN: 2789-6838

Load balancing techniques in cloud computing: A review

Mohammed Khawwam Ahmed[1, 2*], Salah Awad Salman[2], Omar Younis Abdulhammed [3]
1- Technical Institute of Baqubah, Middle Technical University, Iraq
2- College of Computer Science and Information Technology, Anbar University, Iraq
3- College of Science, University of Garmian, Iraq

https://doi.org/10.54153/sjpas.2024.v6i1.526

Article Information
Received: 14/05/2023
Revised: 25/06/2023
Accepted: 30/06/2023
Published: 30/03/2024

Keywords:
Load balancing,Virtual Machine, cloud computing, QoS.

Corresponding Author
E-mail: moh20c1005@uoanbar.edu.iq
Mobile: 07728827864

Abstract

Cloud computing provides an easy and flexible accessibility of resources on the Internet. In this case the clients can use the available resources as they need without upgrading their own hardware. Thus, load balancing is considered as one of the most challenging issues related to the cloud computing where multiple tasks (processes) must be run simultaneously on the processing elements. There are different algorithms used for the task's allocation on those elements. The tasks can be distributed according to different schemes. Some algorithms suggest prioritizing the tasks while some others distribute the balance according to the length of the task. However, there is a large number of the load balancing methods that depends on the Artificial Intelligence techniques. Specifically, the utilization of the meta-heuristic algorithms for the task's distribution on the virtual machines. The aim of these algorithms is to enhance the cloud system productivity by looking for the optimal distribution of those tasks on the virtual machines. There is a large number of methodologies that is worthy to review and investigate in terms of their efficiency, performance and productivity. The main aim of this work is to make a comprehensive literature review paper that discuss the advancement of this area through the years. The advantages and disadvantages of those methods are investigated in order to highlight the gaps and try to suggest some solutions in future. In addition, the classification is conducted on basis of the parameter coefficients. Besides, a full analysis has been made for the compared methods. The futuristic aspects of the utilization of the currently used load balancing algorithms has also been highlighted.

Introduction

The fast development in the computer science and Internet technologies lead to the appearance of the cloud computing which is a virtual manner to handle the processing of tasks simultaneously which is parallel processing scheme. One more benefit is that clients only pay for the service they need at a time [1].

The cloud computing is a time-sharing system that is considered a novel advancement that is characterized with scalability in service provision. This technology offers an easy clients access to the system which is an on demand service utilization. The client then has the

prosperity of being in a resources pool which is a self-served as well as a scalable domain. In turn, this kind of computing is a green computing because users need no much energy consumed to operate an extra processing hardware. Users only need the power to handle what they really need. Cloud technology will be the normal computing technology for the future generations [2]. The resources usability.

The main objective of using the cloud technology systems is to provide the service to the clients wherever they exist. Therefore, the clients will no longer need to take their computing hardware when they are away. Many computing assets are housed in data centers which represent the cloud environment. Within the last decade the cloud services have been requested in an increasing manner. Therefore, there become a heavy load on the network environment. The heavy pressure on the network represents a challenge in the cloud computing domain. It is a performance declination factor that wastes the assets and time. The resource usability and the Quality of Service (QoS) are considered two major objectives in terms of an optimal task's distribution. Additionally, the run time and execution time are both measurements for an efficient resource's management [3]. In the cloud environment the information system can be expanded via the increment of the system's capacity by the dynamic amalgamation of the software licenses, infrastructures, or the use of new equipment. The easy accessibility to the resources which is customized upon the users' need is considered as one of the advantages of the cloud technology[4].

The cloud offers several things to deliver including SaaS, IaaS, and PaaS all these services are controlled by the cloud technology which are online managed by the main hub server. IaaS offers clients a capability to access to the storage, operating systems and server items of the cloud environment on a according to the users need. A large spectrum of services is offered by the IaaS service. Platform as a Service (PaaS), provides an environment that enables the users creating, testing, hosting, and maintaining their data and processes. A cloud environment service manager can host the applications and make them accessible to the users in an online mode. In the last few years, it was well perceived by different jobs that the utilization of the cloud assets and acquiring an easy accessibility leads to a notable reduction in the costs expected to finalize any specific job by applying the idea of the service-wise payment that suggests the users only pay for the services they need. It is a time-saving principle that makes users not necessarily upgrade their hardware assets. Because of easy implementation, cost effectiveness and the lower investment costs, there is a tendency toward the use of the cloud services by different businesses. Some of the benefits of the cloud technology utilization in businesses is the lower costs, security, fault tolerance besides the excellent performing capabilities. Large number of the big companies such as IBM, Microsoft and the Amazon have adopted the cloud technology in their services provided to the clients[5].

With the fast-technological advancement, there become a large use of online services by users. The services required by the users nowadays an easily provided by the cloud servers. In this case, there is no need to buy those services rather they only be rent while those services are required. The heavy loads on the cloud networks lead to the fact that this load might not be distributed optimally where there is an urgent need for sophisticated tools that enable the network to be configured by using protocols that handles the incoming processes by distributing them efficiently on the virtual processing elements. Still, the tasks distribution and

load balancing issues can be clear in real-times applications where users need the services immediately. However, due to imbalance in the distributions of the other tasks submitted by other users delays or even services denials might potentially occur. Consequently, s system crash will be the last result of the heavy traffic on the services provided by the cloud [6]. The load balancing is considered as one of the gravest issues related to the cloud technology. This process of the optimal distribution of tasks on the processing elements is necessary at the times when any particular component fail to provide it functionality, the system would be capable of de-providing applications on other processing elements automatically. Thus, the load balancing is a technique aims to divide processes fairly on the machines. The idea behind load balancing is to guarantee there is no single machine to be over-loaded while some other are idle or less preforming. The user satisfaction can be acquired with the assistance of using the load balance mechanisms.

Scalability in cloud computing, facilitated by load balancing, is another aspect that might be considered. These aspects collectively enhance green computing by promoting resource efficiency and reducing carbon emissions.

The improvement of the general system performance comes through the optimal Resource Allocation (RA). The key issues in the cloud environment are the assets consumption besides the energy conservations. In this case, the works become more environmentally friendly through lowering the cost and reducing the processing time. Load balancing can be described as a method that spread the workloads on the cloud environment for the sake of maximizing the throughput and reducing the feedback time. The process of load balancing incorporates the preventions of the assets from being malfunction due to overloading. The dataset could be received and transferred with a minimal delay amounts because of the traffic distribution on the servers. The dynamically based divisions of workloads are a crucial parameter in cloud technology. Machines work load is the total quantity of processing intervals required for the completion of the processes assigned to the system. The necessity of using the load balancing is to reduce the heavy loads on the processors with keeping the system load balanced as much as possible.

The benefits of using this technique is to maximize the resources usability. In turn, the system performance will be maximized accordingly. By achieving this the cloud system will gain the users satisfaction. Thus, the maximization of the throughput besides the minimization of the feedback time are the two factors that the cloud system must achieve in load balancing. Thus, load balancing is considered as crucial technique that is used to enhance the online services and operations performance[7].

Based on the information provided in [8,9] the key outcomes gained by the load balancing can be listed bellow

- Efficiency- the bandwidth utilization can be increased in the network.
- Safety and Reliability- dependability and security can be enhanced
- Scalability- capacity for growth can be maximized
- On-demand assets accessibility.
- Efficient utilization of resources regardless the traffic.
- Energy can be saved during the low loads.

- Cost effective
- Minimal processing time.
- Elimination of assets wastages.
- Better communicating of network nodes.

Hardware and software in two primary categories are used in server load balancing. Hardware that operates at the 4th layer, such as network switch, provides load balancing (LB) and server redundancy. Still, software is used to manage resource allocation and track server utilization. Statically and dynamically based load balancing solutions are distinguished based on whether software or hardware is used. While dynamic techniques adjust to changing demands, static methods are used proactively to handle projected processing loads. Optimizing resource utilization, raising throughputs, guaranteeing quicker response times, and avoiding overload are the main goals of load balancing. Diverse methods are applied to disperse the load among various systems in order to handle various issues. Every load balancing algorithm is designed to accomplish specific goals. For instance, some of them they strive to achieve higher throughputs while some other methods aim to achieve a reduced response time, and efficient resources usage [10].

Motivation

This paper aims to offer the most recent advances in the area of the task's distributions method. In this paper, a guideline for researchers is presented in order to highlight the research gaps as well as the strength points in the load balancing related academic works.

- A comprehensive literature review is presented in this paper focusing on the work load distributions on the virtual machines in the cloud environment. This paper involves reviewing the current research ideas in order to obtain the best work load management among the virtual units.
- A load balancer ensures fair allocation of resources and maximises customer happiness while minimising costs.
- Load balancing assist in guaranteeing users to be satisfied by receiving a cost-efficient service besides a fast service.
- Load balancing among cloud resources is important because of the increasing need for the cloud computing

Literature Review

In this section a survey is done in concern to the load balance in the cloud environment besides other considerations associated to this subject. For instance, the resource schedule, services broker rule, and the assets assignments.

Through the years there are several works has been made in the areas of the cloud technology. In Jiang, Y. (2015). Investigated the ideal characteristics discovered in a wide spectrum of distributed system. This can be classified into a distributed controller, open environment, resources assignments, varying node units and the networked architectures. The paper tends to categorize the studies made on the task's allocations and load balancing according to the users' satisfaction. The criteria considered in the work are the ensuring, the reliability, optimization of using the resources, the coordination of the mechanisms, and the

accountability of the network's architecture. The paper presented a full analysis related to the job distributions[11].

Singh, S., & Chana, I. (2016) conduct a thorough and precise analysis of the current research on managing cloud resources, particularly emphasising the scheduling of these resources.
The major aim is to choose the highly significant and suitable technique with large spectrum of assets allocation algorithms for specific works loadings. The aim of the work is generating an extensive methodologically based analyses of the network resources management and plans. The investigations involved different principles like the resource's classifications. The abundance of various techniques. The scheduler techniques are distributed according to specific strategies and considering the considerations scalabilities. Prior to conducting comprehensive resource planning analyses, it was advised to improve the current cloud search capability. It was discovered that resource utilisation might be enhanced by assigning resources based on the characteristics of the task. Furthermore, they provided recommendations for forthcoming endeavours [12]. The paper was authored by Milani, A. S. et al. (2016). Offers a thorough analysis of the load balancing approaches being utilised. In addition, a comprehensive classification has been presented, considering many aspects, along with an analysis of the current methodologies.

A study has been conducted to analyse the advantages, disadvantages, and important challenges that are associated with a variety of load balancing algorithms [13]. The purpose of this analysis is to enhance the performance of load balancing systems in the future. Xu, M., Tian, W., and Buyya, R. are considered to be the authors of the publication. 2017 was the year that the publication was released. This article provides a comprehensive and comparative study of the most recent research on load balancing strategies for virtual machines in cloud computing. The primary focus of the research is to investigate the features of various virtual machine load balancing components. These components include scheduling scenarios, management strategies, resource types, consistency of VM types, and allocation dynamics. After some time had passed, the parameters that were utilised for the purpose of planning load balancing of virtual machines were compressed in order to assess the effect of load balancing and other scheduling goals [14]. (2017) publication by Thakur, A., and Goraya, M. S. A thorough investigation into the most recent and cutting-edge load balancing techniques for the cloud computing environment was carried out. Employing a novel classification framework for cloud load balancing algorithms in order to achieve this. The assessment includes a clear presentation of a variety of potential load balancing procedures as well as a comparative evaluation of those same approaches. The survey covers a wide range of issues related to the load balancing problem line of action that may be taken to solve them [15].

Hota, A., Mohapatra, S., and Mohanty, S. (2019). Examined several scholarly literatures pertaining to various load balancing methodologies. The LB algorithms were classified into three distinct categories: heuristic, metaheuristic, and hybrid, based on the specific technique employed. In their poll, they conducted a comparison of all three options and obtained the following results: (1) Heuristic algorithms are characterised by their ability to quickly generate a satisfactory solution, in contrast to metaheuristic algorithms; (2). The evaluation of performance for metaheuristic algorithms is contingent upon the characteristics of the problem, its inherent structure, and the employed solving methodology; and (3) Hybrid algorithms offer the advantage of reducing both computation time and cost.

It shows an advanced effectiveness in comparison to the rest of the techniques [16]. The writers in are Jyoti, A., Shrimali, M., et al. (2019). This research offered a full an extensive investigation of many processes' distribution technique broker items, scheduler kinds, and tasks the time 2015-2018. The authors investigated as well as investigated the modern technologies with an extensive argumentation about this subject [1].

The load balancing approaches were categorised and an overview of the persisting problems and challenges was provided [17]. Elmagzoub, M. A., Syed, D. et al. (2021). Provides an overview of LB algorithms that draw inspiration from swarm intelligence (SI). The SI include a wide spectrum of method that mimic the synergy of the social behaviours in biology. There are a lot of examples such as the particle swarm optimization and other methods. The fundamental objectives, application scope, and specific issues addressed by each algorithm are thoroughly analysed, along with any enhancements made. In addition, an evaluation of performance was carried out, considering mean response times, data centre processing times, and additional quality indicators [18].

Load balancing Challenging issues

The Clouds technology is now receiving the greatest attention in the area. Cloud computing research is currently grappling of the problems associated with this subject among many other challenges. In order to determine the most effective method for improving the utilisation of cloud resources, it is necessary to address several additional concerns, like virtualized machines (VMs) migrations, VMs securities, user's qualities of services (QoS) satisfactions, and asset usage, with equal importance [1]. The following is a compilation of many LB challenges:

- **Geographically based Distributing Node**: Clouds information centres are scattered across several locations for computational purposes. These centres consider spatially distributed nodes as a unified location system to efficiently carry out user requests. Due to the limited scope of certain load balancing strategies, factors such as network latency, communication delay, and the geographical distance between distributed computing nodes, as well as between users and resources, were not considered. Consequently, the algorithms are ill-equipped to handle nodes that are located far apart in this particular environment. Therefore, it is essential to consider creating load balancing algorithms for nodes that are situated at a considerable distance from each other.
- **Single Point of Failures:** Several dynamically based loads balance (LB) techniques have been developed. A number of them utilise non-distributed methods and depend on the central node for loading balances choices. of the event of a failure of the primary device, the entire computer ecosystem will have adverse consequences. Hence, it is imperative to develop distributed algorithms that ensure no individual node has complete control over the entire computational system.
- **Virtual Machine Migration:** Virtualization refers to the procedure of establishing or consolidating several virtual machines (VMs) on a single physical system. The deployed virtual machines will exhibit distinct behaviour due to their unique setups. During system overload, certain virtual machines (VMs) may need to be relocated using a load-balancing method for VM migration [19].

- **Heterogeneous Nodes:** Cloud technology necessitates the execution of user needs on nodes with diverse characteristics to optimise resource utilisation and minimise response time. Consequently, scientists have a difficulty in creating efficient load-balancing techniques for the diverse environment.
- **Storage Management:** Cloud storage has effectively addressed the issues associated with earlier conventional storage systems, which required human supervision and incurred substantial hardware expenses. Nevertheless, users may securely keep a wide range of data in the cloud without any concerns about access limitations. The proliferation of cloud storage necessitates the replication of data to ensure rapid accessibility and consistency. Full data replication systems are not highly efficient due to the redundant data storage policy of replication sites. While partial replication may be enough, it might increase the intricacy of load-balancing algorithms and pose a challenge to dataset availability. Implementing an effective load balancing technique is essential. This strategy should be using a partial replication mechanism that takes into account the dispersion of apps and their corresponding data.
- **Load Balancing Scalability:** The instant access and flexibility to increase in size of cloud services allow consumers to easily adjust their usage by accessing services as needed. In order to adequately adapt to these modifications, a proficient load balancer must consider swift fluctuations in compute capacity, storage, system architecture, and other relevant elements.[20] .
- **Algorithm Complexity:** Algorithms employed in cloud computing must consistently be uncomplicated and user-friendly. As the algorithm complexity increases, the performance and efficiency in a cloud environment decrease.
- **Automated Service Provisioning:** Flexibility is the main characteristic of cloud computing, since it allows for the automated assignment or delivery of services. Subsequently, we would have the capability to employ the suitable resources and use or free up cloud resources while maintaining the effectiveness of older method[21] .
- **Energy Management:** The advantages of utilizing cloud-based energy management systems include the ability to achieve scale economies. The paramount aspect in establishing a global economy that relies on a limited number of providers to support the pool of resources, as opposed to individual private services, is energy conservation. [21] .

Difficulties associated with load balancing in cloud computing

Cloud computing is currently garnering significant attention in the field. Cloud computing research is presently addressing the topic of load balancing, along with several other obstacles. To identify the optimal approach for enhancing the utilization of cloud resources, it is imperative to consider various additional factors of equal significance. These include virtual machine (VM) migration, VM security, user quality of service (QoS) satisfaction, and resource utilization [1]. Below is an aggregation of several LB challenges:

- Geographically distributed nodes: Cloud data centers are strategically distributed across many locations to optimize computing capabilities. These centers utilise a network of scattered nodes to effectively process user requests by treating them as a unified

location system. The load balancing solutions did not consider aspects such as network latency, communication delay, geographical distance between distributed computing nodes, and distance between users and resources due to their restricted scope. As a result, the methods are not well-suited to manage nodes that are situated at significant distances from one other in this specific environment. Hence, it is crucial to contemplate the development of load balancing algorithms for nodes that are located at a significant distance from one another.

- Single point of failure: Various dynamic load balancing (LB) algorithms have been developed, with some using non-distributed methods and depending on a single node to handle load balancing choices. In the case of a primary device failure, the whole computer ecosystem will experience detrimental implications. Therefore, it is crucial to create distributed algorithms that guarantee no one node possesses absolute authority over the whole computing system.

- Migration of Virtual Machines: Virtualization is the processes of creating using many VMs on a stand-alone physical machine. The used virtual machines had diverse behavior as a result of their individual setups. In the case of having an overload on the physical machine there will be a need to move the processes, employing a load-balancing technique for virtual machine migration [19].
- Heterogeneous Nodes: Refers to nodes in cloud computing that are diverse in nature., it is necessary to perform user tasks on nodes that have different characteristics in order to maximize the usage of resources and decrease reaction time. Hence, scientists have challenges in developing effective load-balancing systems for the varied environment.
- Management of storage: Cloud storage has successfully resolved the challenges associated with previous traditional storage systems, which need human oversight and entailed significant hardware costs. However, users may confidently store a diverse array of data on the cloud without any worries about restrictions on access. The widespread use of cloud storage requires the duplication of data to guarantee quick availability and uniformity. Data replication systems suffer from low efficiency as a result of the redundant data storage strategy implemented by replication locations. Although partial replication may suffice, it might complicate load-balancing techniques and present a hurdle to dataset availability. It is crucial to use a proficient load balancing approach. The approach should be founded upon a partial replication system that considers the dispersion of applications and their corresponding data. Cloud services provide users the flexibility to easily modify their consumption by accessing services as required, thanks to the on-demand accessibility and scalability of load balancers. A sophisticated process scheduler must be used in order to efficiently adopt to these changes where a special attention should be given to the computational powers, storages as well as the fluctuations that might be happening [20].
- Algorithms Complexities: Algorithm utilized in the network must continually exhibit simplicity and ease of usage. As the algorithm's complexity rises, the performance and efficiency in a cloud environment diminish.
- Automation in Services Provision: The primary attribute of cloud technology is its flexibilities, since it enables the automated allocation or distribution of services. As a result, we would be able to use the suitable assets and allocate or free up clouds resource

whereas maintenance traditional models. [21]. Cloud-based energy management solutions have the benefit of achieving scale savings in energy management. The key factor in creating a worldwide economy that depends on a small number of suppliers to sustain the pool of resources, rather than individual private services, is energy preservation [21].

Load Balancing Metrics

This section focuses on the measures used to evaluate tasks distributions. For enhancing resource utilization and efficiency, it is important to employ a load balancer to evenly spread the computational workload among the accessible resources. Researchers have proposed a range of load balancing (LB) approaches and associated measures to enhance customer happiness and optimize resource use. These measures have been the subject of debate by[8], [16], [22 − 28], as follows:

- Performance: This evaluates the effectiveness of the system after implementing the LB technique in comparison to other LB methods currently in use.
- The response time is a measurement that determines the total amount of time required to fulfil a request that was supplied by the system.
- The term "throughput" refers to a measurement that measures the number of operations that are finished within a specific amount of time. When the throughput of the system is raised, the system demonstrates improved performance.
- When referring to an algorithm, scalability is the capability of the method to spread the workload in a system in an even manner as the number of nodes rises.

- The term "makespan" refers to the process of determining the maximum amount of time that a user will need to finish a job or access resources in order to be considered successful.
- The term "fault tolerance" refers to the capability of the algorithm to keep its performance consistent and accurate, even in the event that there is a failure at any link or node within the system.
- "Migration Time" is the amount of time that must pass before a request or work may be moved from a machine that is under a significant amount of load to a machine that is under a lower amount of load. A decrease in the amount of time required for migration improves the performance of the cloud system.
- Resource Utilisation: This indicator evaluates how well all of the available resources in a cloud environment are being utilised while maintaining effectiveness. In proportion to the growing utilisation of information technology resources, the overall cost, energy expenditure, and carbon emissions are all decreasing.
- Imbalance Degree can be used to quantify the discrepancy that exists between different virtual machines.
- It determines the quantity of energy that is spent by each node, which is referred to as energy consumption.

Load Balancing Classification Techniques

There are several academic works such as those mentioned in [1], [29], state that the categorization of load balancing strategies mostly revolves on the state of the systems and the process of process start. Load balancing techniques that utilise system state can be classed as either static or dynamic, as seen in Figure 1. Such techniques according to their initiation party i.e. reliever, sender or system based initiation.

Process Initiation Techniques:

- Sender Initiated: This strategy involves a node proactively searching for other nodes that are less burdened and capable of helping to distribute the workload when it becomes overburdened. Upon detecting overloaded nodes, the sender initiates a search for lighter loaded nodes.
- Receiver Initiated: This technique entails less occupied node that include receivers, proactively seeking highly jammed nodes of offload the burden. The goal is to equally distribute the workload among nodes in a fairer manner.
- The symmetrically based technique integrates sender-initialized as well as receiver initialized process approaches.

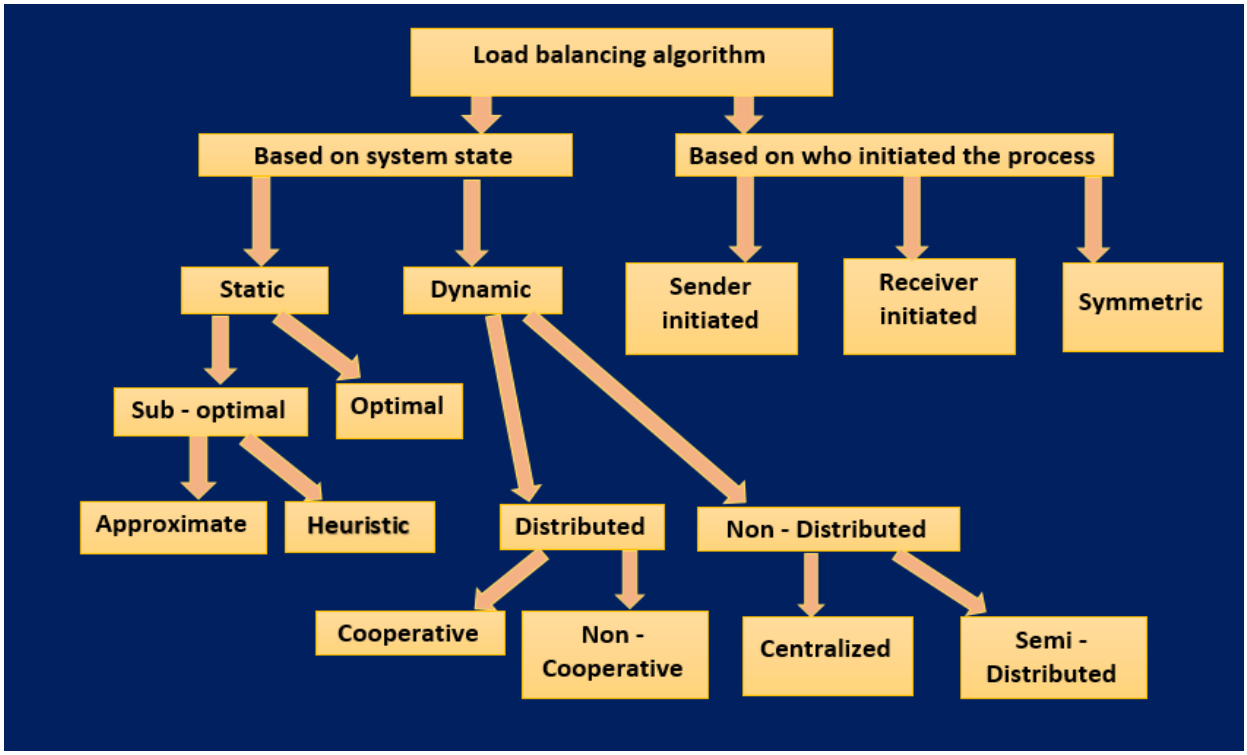

Fig 1: Modern categories of LB methods.

Based on system state techniques:

- **Static**: Static load balancing solutions adhere to a pre-established set of principles that are not influenced by the current condition of the system. Due to the inflexible nature of static algorithms, it is imperative to have prior knowledge of the memory and storage capacity of each node, as well as the computing capability of each node. This strategy is straightforward and user-friendly, but it often fails to detect the connected servers, resulting in an imbalanced allocation of resources[8]. The static load in cloud computing solutions is often predicated on two assumptions. The first factor is the reception of the

original assignment, while the second factor is the prompt accessibility of tangible apparatus. The resource update will occur after the scheduling of each job. The primary issue with this approach is the absence of consideration for the system's present condition while making decisions. Consequently, it is unsuitable for distributed systems that undergo dynamic changes. Static approaches function optimally only when the nodes encounter little load variability. These algorithms are straightforward and have minimal cost in terms of implementation. It excels in uniform surroundings with fast communication speeds when communication delays are disregarded[30]. These algorithms do not consider the continuous monitoring of network nodes[13]. The references [9], [31], and [32] provide evidence of the fact that a number of studies have been carried out. Due to the fact that the method did not have fault tolerance, it was not feasible to implement it in a cloud environment that is dynamic. According to the information presented in sources [1] and [33], static load balancing strategies may be classified into the following categories: The Data Centre Controller is responsible for gathering information about the available resources and delegating tasks to the load balancer. The load balancer successfully distributes the tasks in the shortest period of time possible by employing the most efficient methods. Ineffective: If the load balancer is unable to decide the most effective course of action, then it will generate a solution that is ineffective. There are many different scheduling algorithms that depend on the length and size of each single process, some of the static load balancing strategies that are often utilised in virtual machine settings. There are several properties that define a static algorithm:

1. They arrive at a choice by using a predetermined criterion, such as the input load.
2. They have a deficiency in adaptation.
3. A prerequisite is a familiarity with the system.

- **Dynamic** In the realm of cloud computing, this method is of utmost importance. These algorithms make judgements based on the present state of the system, taking into consideration how loads are distributed across the operating physical machines. The primary advantages of these approaches are the ability to transfer duties from a heavily burdened machine to a less burdened machine. Dynamic load balancing approaches enhance system performance by providing increased flexibility. A dynamic method does the following activities while processing: It consistently monitors the workload of the nodes. The system computes the workload of each node at regular time intervals and redistributes the task across the nodes by sharing information about the load and status. An overloaded node redistributes its load to an underloaded node during execution, dynamic algorithms modify the distribution of workloads across nodes. Based on the current knowledge about the VM's capabilities to hasten Subsequent distribution decisions involve the allocation of workloads as described in reference 28. It efficiently decreases communication delays, processing time, and execution, enhancing its flexibility in adapting to alterations in cloud configurations [9]. As stated by Mishra, S.K et al. (2020).The dynamic load balancing techniques based on heuristics may be categorised into two groups: batch mode (offline mode) and online mode. The

assignment of the work is only based on batch mode heuristics, which are implemented at specific specified intervals. It is utilised to approximate the real duration required to finish a larger quantity of jobs. Various strategies, such as Max-min, Min-min, and the Suffrage Algorithm, have been proposed for batch modes. Upon arrival at the scheduler, a user request (task) is promptly assigned to a computational node in real-time mode (sometimes referred to as instant mode). Given that each work is scheduled just once in this scenario, the scheduling result remains unchanged. OLB, MET, MCT, and SA are among the heuristics proposed for online modalities [30].

- These algorithms are well acknowledged for their exceptional efficacy and flexibility in load distribution. Unlike SLB algorithms, load balancing algorithms in a dynamic environment consider the previous state of the system [34]. Although these algorithms may seem difficult and impose a significant system overhead, their main advantage lies in their adaptability. There are many different meta-heuristic methods such as the swarm intelligence methods as well as the evolutionary algorithms. These methods are all iterative based methods that optimize a specific condition [35-37]. The categorization of additional techniques for spreading labour in a dynamic way is as follows:

- **Distributed:** All nodes actively participate in load distribution using distributed approaches such as job scheduling or resource allocation [38]. Every node maintains a communication information base to facilitate effective task allocation and reallocation. Cooperative or non-cooperative distributed algorithms have the ability to communicate with each other. Cooperative systems include the collaboration of all nodes to achieve a shared goal or make choices. If this collaboration is absent, the system is considered non-cooperative.
- **Non-distributed:** As stated by Kumar, P., & Kumar, R. (2019). In centralised approaches, the decision of load distribution is decided by one or more nodes. Undistributed approaches have the ability to function in a centralised or semi-centralized way. A singular node is employed in centralised techniques. Manages load balancing and oversees all load distribution operations. In the case of a single-node failure, centralised approaches might result in the irreversible loss of node information. Nodes aggregate to create clusters, which function as centralised mechanisms in the semi-distributed method.[1]. The characteristics of dynamic algorithm are:

  1- They make decisions depending on the current condition of the system.

  2- Adaptable

  3- They improve the system's functioning.

LB algorithm policies in a dynamic setting

In addition to what was said earlier, dynamic LB algorithms make advantage of the present condition of the system. It is via the implementation of certain policies that they are able to do this [22], [28], [39-43]. It is these policies that:

- Transferring Policies: This policy outlines specific conditions that warrant the relocation of a task from one node to another. The transfer policy determines the routing of

incoming tasks, deciding if they must be moved or processed locally, depending on a predefined set of rules. The performance of each node has an impact on the functioning of this rule. This guideline is applicable to both the relocation and rescheduling of tasks. The selection policy determines which tasks should be migrated based on many factors, including the task's execution duration which represents the count of the number of times the call is made.

- Location Policy: This policy entails the identification of underutilised nodes and the allocation of jobs to them. The policy also verifies if the targeted node possesses the requisite services for jobs to be moved or postponed. The Information Policy involves the acquisition of data through the transfer policy, which gathers information about the nodes in the system and determines the optimal timing for data gathering. It is the responsibility of the transfer policy to examine incoming tasks and determine whether or not they should be sent to a remote node in order to achieve load balancing. It is necessary to process the task locally if it is not possible to relocate it, and the location policy is activated in order to locate the work if it is required. It is the responsibility of the information policy to provide the transfer and location policies with the data they require to make reasonable decisions.

There are now seven existing load balancing approaches.

We conducted a comprehensive assessment of the LB techniques documented in the literature and thoroughly examined the selected publications. The system state categorises LB methods into static and dynamic types.

Within this part, we shall examine the current methods employed for LB.

Techniques for Load Balancing in a Static Environment

Static load balancing approaches observe system resources such as execution time, memory, processing power, and storage capacity without needing information about the present state of the system. These strategies hinder the allocation of resources during runtime. They are easily implemented and suitable for tiny networks or systems with low resources. Nevertheless, they are ill-suited for distributed computing systems as they fail to consider the current status of the system, leading to an inequitable distribution of resources. Furthermore, they lack the capability to identify the server machine that is connected during runtime. The load balancing mechanism called "dynamic exchange" can be used to disperse a finite load across the machines that are currently available. The strategy of employing static load balancing on processors with a predetermined workload and synchronous communication entails the allocation of tasks into individual units known as tokens, which are of uniform size, utilising only local computing. This is executed within a model that incorporates a solitary port and a solitary token. The burden remained unchanged until it was reallocated. Processors operating in a dispersed environment with dynamically changing loads found it to be ineffective. A static load balancing approach ensures that all task-related information is accessible prior to minimising the waiting time for tasks. It calculates the projected duration for completing each job from a given set of incoming tasks. Jobs with the shortest durations are prioritised and handled first, while jobs with the greatest durations are processed last. Certain tasks may experience hunger as a result of increased processing time demands. Static task-based load balancing enables the implementation of round-robin task execution [45].

The load-balancing approach is employed to handle many jobs, with each work being executed for a specific time interval before being placed back in the queue. While the uniformity of requests makes it advantageous for web servers, it lacks use in cloud settings where processes may have varying configurations. The value is represented by the matrix element at reference[46].

Dynamic LB Techniques

They Static approaches are unsuitable for distributed computing systems that undergo dynamic state changes and require real-time information on resource states. Therefore, knowledge of the system's present state is not necessary. Therefore, we need dynamic load-balancing strategies that are appropriate for cloud systems. In this part, we discuss different dynamic load balancing approaches that depend on load balancer criteria [1]. Based on our observations, we have classified them into the following categories:

- Overall Load Balancing
- Load Balancing based on Natural Phenomena
- Combination Load Balancing
- Tasks distribution using Agents
- Tasks distribution or load balancing
- Tasks distribution within Clusters

Load Balancing Strategies: We explore several factors and examine many generic strategies, such as VM migration and load estimate algorithms. As stated by Bala, A., & Chana, I. (2016). Utilising machine learning (ML) is advisable for the purpose of forecasting node load in order to effectively administer it inside the cloud environment. A load-prediction model utilising Random Forest was created in the Cloudsim tool environment to enhance load balancing in VMs. The model underwent testing just in a simulated environment, but it has the potential to be evaluated in real-world scenarios by including fault-tolerance mechanisms.

Ghoneem, M., and Kulkarni, L. (2017). The active load-balancing approach of a cloud analyst simulator's VM has been modified. The existing virtual machine load-balancing technique exhibits an unequal distribution of requests, resulting in a longer response time for consumers, particularly during peak hours. The allocation and reservation tables were utilised to monitor VM reservations for the requests. Upon considering the load balancer's data, the data centre allocates a virtual machine (VM) for the upcoming user request. The findings demonstrate that, even during periods of heavy congestion, the recommended method provides the fastest response time to customer inquiries. The value is represented by the matrix [48].

To alleviate the burden on servers, they used an innovative load-balancing strategy instead of a static technique. This new strategy prioritises server processing power and computer loading. The authors analyse the different load balancing measures utilised by the existing implemented methods and compare them with the approach being evaluated. Despite the neglect of migration and reaction times of jobs, these characteristics might be integrated into the current ones to enhance the load balancing algorithm.The value of the expression is 8. Table 1 presents a comprehensive examination and summary of many common load-balancing methodologies.

Table 1: Synopsis of General LB Techniques

References	System State	Technique	Concept	Pros	Cons
[**47**]	Dynamic	An algorithmic load-balancing approach that leverages machine	Employing machine learning techniques to detect instances of excessive load and insufficient load in a machine	Optimal utilisation of resources, little impact on migration process, and reduced frequency of migrations	It has not undergone testing on an actual cloud infrastructure.
[**48**]	Dynamic	Modified VM load balancing approach that is currently in use	Utilise a reservation table to guarantee equitable distribution of requests.	Optimising task reaction time, distributing workload evenly among virtual machines, and enhancing scalability	Tasks are evenly allocated throughout a solitary data centre.
[**8**]	Dynamic	An unique load balancing technique aimed at minimising server demand	The term "dynamic" refers to something that is characterised by constant change, activity, or progress.	increased resource usage makespan, as well as QoS	The process of annexation is favoured over static load balancing. Increased use of resources

Natural Phenomena Inspired LB Techniques:

Table 2 presents a concise overview of load balancing techniques that draw inspiration from various natural occurrences. There are a lot of natural examples that copy the real-time of species to perform the optimization. As stated by De Falco, I. et al. (2015). A strategy based on extremal optimisation (EO) has been developed to distribute the burden of user requests that are executing simultaneously. A population-based parallel evolutionary optimisation approach incorporates factors such as target node selection and fitness function, which yield solutions for dynamic methods [49].

Load-balancing strategies are necessary to enhance resource utilisation, minimise response time, and expedite job completion for cloud-based operations. Remesh Babu, K. R., and Samuel, P. (2016). A approach has been developed that use load-balancing characteristics such as response time, quality of service (QoS), and migration volume. The food supply for the honey bees are the underutilised virtual machines (VMs). When a virtual computer got overwhelmed, certain processes with low priority were moved to another virtual machine. Additional nature-inspired algorithms, the most remarkable methods in this domain is using the Swarm Intelligence Optimization methods. might be included into the algorithm to enhance its performance [50].

In order to enhance the speed at which tasks are executed, we propose an enhanced weighted round-robin algorithm that takes into account the size of the jobs, the capacity of the virtual machines, and the interdisciplinary of several activities. The algorithm selects the machine with the shortest completion time by taking into account the current workload of all virtual machines and the time it takes to execute the operation. Upon completion of each job, a load balancer is initiated to evenly divide the workload across the nodes. To enhance the algorithm, it is possible to utilise machines from different surroundings in a heterogeneous setting, which would yield consistent outcomes [51]. Table 2 presents an analysis of load balancing approaches that are impacted by natural occurrences.

Table 2: Synopses Real-life Phenomenon mimicked to be used in load balancing

Reference	System States	Techniques	Concepts	advantages	Disadvantage
[**49**]	Dynamic	Load balancing is performed dynamically based on extremal optimisation. Implementation of concurrent tasks in a changing environment	Implementation of concurrent tasks in a changing environment	Decreased execution time, minimised task transfers, and enhanced resource allocation efficiency	The concepts of multi-objective optimisation and graph optimisation are completely disregarded.
[**50**]	Dynamic	Improved bee colony based LB	The honey bee approach is employed to minimise resource consumption and reduce reaction time.	Shorter reaction time, increased resource utilisation, and reduced task migrations	The potential for expansion and the level of intricacy are restricted.
[**51**]	Dynamic	Weighted round-robin	Considering the duration of	Reduce response	Execution in a homogeneou

	technique	task execution, the reaction time of the job is reduced.	time	s environment

Hybrid LB Techniques:

Hybrid load balancing solutions aim to surpass the limitations of both static and dynamic load balancing systems while preserving their own advantages and characteristics. Hybrid methodologies enhance response time and optimise resource use.

A hybrid load balancing system is employed to optimise node management and improve efficiency. This system integrates features such as scheduling, querying, and moving jobs on demand, along with staged task transfer [52]. If a node identifies a significant workload, it will redirect the most recently arriving node to a less congested one. Both the QMT and SMT exhibit equivalent efficacy for both autonomous and interdependent tasks. Nevertheless, they encounter extended periods of transmission and scheduling lengths, which might potentially be alleviated in the future. The authors of the study are Naha, R. K., and Othman, M. The current year is 2016. Have developed a hybrid algorithm that combines many approaches to get optimal system performance. According to the authors, there are algorithms that are designed to enhance resource utilisation by being both Load Aware and Cost Aware. The Cost Aware algorithm focuses on minimising expenses, while the Load Aware algorithm prioritises faster processing time even if it results in higher costs. When a customer who is conscious of costs sends a request to a server, they disable the fast (high-cost) backend, leading to decreased resource consumption and longer processing time for user queries. The service-broker algorithm selects the server according on the specific needs of the client, perhaps resulting in longer processing times or elevated expenses. The service proximity broker algorithm selects the data centre in closest proximity to the customer's location. The results provided above demonstrate a decrease in both computational and reaction times. In order to fulfil the customer's requirement, the authors included all of the load balancing and service-broker algorithms within the pair. Nevertheless, the extended period of request execution has the potential to enhance the system's performance.

In order to reduce the possibility of server response failure due to a sudden increase in user requests, it is crucial to develop a strong and reliable cloud load balancing (CLB) architecture. The authors' depiction of this architecture presents a load balancing strategy that efficiently manages server load, priority, and processing capabilities for both physical and virtual web servers. In addition, it considers server processing and computer loading to address server problems and handle a larger volume of computation requests. The architecture enables performance scalability, but it also results in increased response time (table 3).

.

Table 3: Synopsis of Hybrid LB Techniques

References	System State	Technique	Concept	Pros	Cons
[52]	Dynamic	Hybrid LB as well as Tasks scheduling	LB on slaves node with hybridized LB based on the master node	Task scheduling for independent and dependent tasks, reduced response time	High transmission and scheduling time
[53]	Dynamic	Combination of broker and LB technique	Using a combination of LB and a broker technique to reduce response time	reduced processing and response times	Low efficiency, lengthy execution
[8]	Dynamic	Cloud load balancing technique	A load balancer monitors priority and computing power to overcome server response failure.	High scalability	High response time

Agent Based LB Techniques:

An analysis is made to investigate the performance of the agents in this realm. Many agent-based tasks distribution strategies has been discussed extensively in this section. Those strategies aim to optimally distribute the tasks on the cloud environment on an efficient manner that is based on the agent which can be considered as a controller that schedule the tasks and then disseminate them over the virtual machines on the environment. The agent is responsible for the discovery, negotiation, establishment, and administration of cloud resources. In order to accomplish the design objective, the agent functions autonomously and consistently.

Application-aware load-balancing architecture, utilising several agents, facilitates the process of automatic resource provisioning. Tasquier (2015) proposes an architectural design that employs three distinct agents: executors, provisioners, and monitors. Each agent has a specific role, with the executor agent responsible for managing running applications, the provisioner agent handling resource scaling, and the monitor agent monitoring resource

overload and underload situations. Moreover, the suggested approach has the capability to evaluate the present condition of cloud and resource elasticity [54].

The authors of the publication are GutierrezGarcia, J. O., and Ramirez-Nafarrate, A. (2015). Present a collaborative agent-based load balancing solution that distributes workload among servers of different capabilities, employing the concept of live virtual machine migration. In addition, they suggest an agent-based load balancing system that incorporates a load balancing programme for the purpose of selecting the initial host for virtual machines [55]. The multiagent-based load balancing approach enables improved resource utilisation. It utilises techniques from both the sender and receiver to minimise waiting periods and uphold service level agreements (SLAs). The DcM agent utilises data from the VMM agent to execute information policy and classify virtual machines (VMs) based on different criteria. Additionally, it initiates NA agents to determine the status of the virtual machines (VMs) that are reachable in other data centres. Based on the simulation findings, it is evident that the approach is more efficient, leading to improvements in both reaction time and makespan [56]. The load-balancing approaches that depend on agents are examined in Table 4. In the papers reviewed in the table it can be realized that the system state, technique, concept besides the advantages and the disadvantages are taken into account in order to show the performance capabilities of each single agent presented in the explanation.

.

Table 4: Synopsis of Agent Based LB Technique

References	System State	Technique	Concept	Pros	Cons
[**54**]	Dynamic	Multiagent-based	Resource provisioning and monitoring in a multi-cloud system involve the use of various agents.	Offers maximum elasticity and makes use of multiple cloud resources.	QoS is disregarded due to its absence of implementation
[**55**]	Dynamic	Agent-based LB	VM migration using agent	Heterogeneous VM and server	QoS is disregarded due to its absence of implementation
[**56**]	Dynamic	Multiagent based LB architecture	Maximizing resource use with multiple agents	diminished feedback time , improved makespan, and improved resource utilization	Data Centre Monitor Agent. This kind of system has a high tendency to be dependant on higher authority managers that is in charge to send

	a message to destruct the data collection whenever required.

Task Based LB Algorithms:

We are now examining the existing literature on task-based language learning (TBLL) strategies in the field of language acquisition. Presently, the prevailing load-balancing techniques entail the migration of an overburdened virtual machine from one physical computer to another. Although this strategy allows for flexibility in distributing resources, it is more laborious and costlier since it requires migrating the complete virtual machine rather than just the individual processes responsible for overloading the system.

Shen, H. et al. (2016). One strategy for optimising the placement of MapReduce jobs is to consider the network topology. By taking into consideration the network, this technique aims to enhance job completion speed while reducing data cost and transmission time. According to them, the primary difficulties of a work are the constant fluctuations in the availability of resources, caused by their release and the frequency of access over time. (2) The duration required to get data for decreased jobs may vary based on their magnitude and location; and (3) The latency of data access is affected by the level of traffic on the pathway. In order to minimise the delay in accessing data, it is important to take into account the load on the path while scheduling jobs. The findings indicate that there was an escalation in resource utilisation, accompanied by a reduction in the time required to complete tasks. Implementing a multiple-scheduler architecture simplifies the process of executing highly parallel jobs and also reduces the associated challenges and expenses.

The authors of the study are Xin, Y. et al. (2017). Proposed a scheduling methodology that use weighted random scheduling to minimise conflicts between tasks and alleviate device overload. Tasks are allocated weights depending on many criteria, such as the time it takes to complete them, the cost involved, and the delay in communication. Machines with lower costs are more likely to be allocated a job. The researchers used a Workflow generator in MATLAB 2012b to create the necessary dataset for their investigations. The dataset included many activities, each with execution time, cost, and transmission delay falling within specific ranges. The study also analyzed the process's structure, scheduling, reliance on devices, and the test conducted on the device set. The results suggest that the utilisation of multiple schedulers enhanced aspects such as device task rivalry and implementation expenses. Nevertheless, the study failed to consider the best values for the parameters, which have the potential to be enhanced [58]. Elmougy, S. et al. (2017). Created a task load balancing approach that merges the advantages of round-robin scheduling with the shortest job first scheduling strategy. In order to equalise the waiting durations for jobs, the system keeps short and long tasks in different ready queues and utilises a dynamic task quantum. The writers have considered both the rate of data transfer and the prevention of resource deprivation. The method was tested extensively using the CloudSim simulator, and the results indicate a decrease in turnaround time, waiting time, and reaction time. The occurrence of prolonged work deprivation has also

been diminished. Nevertheless, the task quantum is not efficient in achieving a balance between jobs that can be optimised in the future to minimise wait periods and the possibility of famine. We examine task-oriented load balancing strategies in Table 5.

Table 5: outlines the summary of Task-Based LB Techniques.

References	Systems States	Strategy	Concepts	advantages	Disadvantages
[**57**]	Dynamic	The architecture of the networks as well as the transmitting period is considered in scheduling	implementing task scheduling in MapReduce to reduce the information sharing overhead	Improved cluster utilization, reduced job completion time	Bandwidth is not allocated, and the mode is not analyzed across different network conditions.
[**58**]	Dynamic	Cost efficient multiple schedulers for parallel tasks	Several schedulers for concurrent job execution, together with a weighted machine allocation technique.	improved resource utilization, task reductionweighting and execution time	The assessment of parameters is not optimal.
[**59**]	Dynamic	An method for task scheduling that combines shortest job and round robin principles, with a dynamic task quantum.	To avoid starvation, it is advisable to segregate small and lengthy jobs into separate ready queues and run them autonomously.	Task starvation and waiting time decreased, while reaction and turnaround speeds enhanced.	Task quantum is less effective

Cluster-Based Load-Balancing Techniques: Resources are allocated across several data centres and organised in a diverse cloud environment, considering factors such as server performance and storage capacity. In this chapter, we examine many strategies that are based on clustering and present a concise overview in Table 6.

To overcome the challenges posed by existing load-balancing techniques, Zhao, J. et al. (2015). We introduced a heuristic approach for LB, utilising Bayes and Clustering, which we refer to as LB-BC. This method has successfully achieved load balancing over an extended period and is founded on the principles of the Bayes theorem. The algorithm calculates the likelihood of physical hosts after an event and includes the notion of clustering to choose the

most suitable host. Factors such as the standard deviation, load-balancing impact, and number of requested jobs are all considered. Subsequently, dynamic load balancing was employed as a benchmark to evaluate the method, since it effectively reduces the standard deviation over a period of time. The proposed methodology is initially applicable only to small-scale environments, but it may be optimised for large-scale networks and time-sensitive applications [60-64]. The user's text is a single period.

Proposed a method for efficiently assigning tasks to clusters, enhancing communication between different cloud systems, specifically for load balancing within the realm of dynamic and real-time multimedia streaming. The system chooses to transfer work requests when the queue has more than five pending jobs. completed prior to commencing the job. This approach surpasses ant colonies, WCAP, and HFA in terms of performance, as well as excels in dispatching jobs more frequently and reacting more swiftly. Moreover, by utilising communication tasks instead of compute jobs, this approach may be improved for real-time scenarios where intermediate nodes experience congestion caused by limited resources, as well as to minimise data loss resulting from congestion [65].

Han, Y. et al. (2017). Developed a tasks distribution system that is structed on basis of a layered network of virtual machines. There is a link controlling all the upcoming processes in order to find the best pathway for them. In this design the self-scheduling strategy had been used. The utilization of this strategy helped to increase the system's scalability. The researchers conducted trials on four distinct computational applications inside a vast cluster and documented improvements in scalability, performance, and reduced communication costs. Future evaluations will assess the approach's suitability for larger clusters and dependencies that involve loops [66]. Table 6 examines load-balancing methodologies that rely on clusters.

Table 6: Synopses of Clustered-dependant Load Balancing methods.

References	System State	Technique	Concept	Pros	Cons
[**60**]	Dynamic	LB based on Bayes and clustering	Clustering is combined with the Posteriori Probability of physical hosts.	The smallest standard deviation and the shortest response time.	Only suitable for LAN use; not suitable for use in a real-time environment.
[**65**]	Dynamic	Cluster based job dispatching technique	expanding massive inter-cloud communication for dynamic and immediate load balancing of multimedia traffic	Better response time	Packet loss as a result of congestion
[**66**]	Dynamic	Self-scheduling techniques for distributed	A distributed self-scheduling scheme that seeks to	Scalability has been improved,	Parallel execution is not permitted.

	models to improve load balancing	enhance LB and scalability.	as has overall performance, and communication overhead has been reduced.

Discussion

These approaches employ several indicators inside the cloud technology for comparing the analysed studies. Through meticulous analysis of various load-balancing approaches, it becomes evident that each method considers distinct criteria for assessment. Some articles have considered either a single target or numerous objectives for measurements.

In addition, we have examined other research publications that depend on simulations and outcomes. There are major concerns that all researcher in this area must be considering such as the makespan time, throughput maximization, besides the energy they might be excessively be consumed in order to solve some small tasks.

Conclusion s

In This study conducted a comprehensive assessment of load balancing strategies employed in cloud settings. In a similar manner, we examined other state-of-the-art load balancing algorithms in cloud computing systems. Our research suggests that load balancing (LB) is a developing mechanism that provides a novel method for improving cloud performance and optimising resource usage. Additionally, LB ensures equitable and efficient allocation of input requests. Upon conducting a thorough examination of the existing literature, we have classified the area into two distinct subdomains, namely dynamic and static LB investigations. We also discussed the advantages and disadvantages of several load-balancing strategies. In order to facilitate the development of more effective LB approaches in the future, it is imperative to tackle the challenges associated with these algorithms. Efficient load balancing can minimise resource usage, leading to additional savings in energy consumption. Overall, LB mechanisms in the computer environment require further enhancement to effectively handle the diversity of the environment, minimise related costs, enhance performance, and really function as an on-demand approach. The collected data from this study provides valuable insights to researchers on the latest advancements in the field of load balancing. The purpose of this study was to provide an overview of load balancing in cloud systems, including its main role, existing challenges, unresolved issues, techniques, and processes. We expect that the findings of this study will contribute to the development of new research areas that will facilitate the advancement and integration of LB in cloud computing.

References

1. Kumar, P., & Kumar, R. (2019). Issues and challenges of load balancing techniques in cloud computing: A survey. *ACM Computing Surveys (CSUR)*, *51*(6), 1-35.
2. Lawanya Shri, M., Ganga Devi, E., Balusamy, B., Kadry, S., Misra, S., & Odusami, M. (2018, December). A fuzzy based hybrid firefly optimization technique for load balancing in cloud datacenters. In *International Conference on Innovations in Bio-Inspired Computing and Applications* (pp. 463-473). Springer, Cham.
3. Velpula, P., & Pamula, R. (2022). EBGO: an optimal load balancing algorithm, a solution for existing tribulation to balance the load efficiently on cloud servers. *Multimedia Tools and Applications*, *81*(24), 34653-34675.
4. Mushtaq, M. F., Akram, U., Khan, I., Khan, S. N., Shahzad, A., & Ullah, A. (2017). Cloud computing environment and security challenges: A review. *International Journal of Advanced Computer Science and Applications*, *8*(10).
5. Chawla, A., & Ghumman, N. S. (2018). Package-based approach for load balancing in cloud computing. In *Big data analytics* (pp. 71-77). Springer, Singapore.
6. Tripathi, A., Shukla, S., & Arora, D. (2018). A hybrid optimization approach for load balancing in cloud computing. In *Advances in Computer and Computational Sciences* (pp. 197-206). Springer, Singapore.
7. Nitin Kumar& Nishchol Mishra(2015). Load Balancing Techniques: Need, Objectives and Major Challenges in Cloud Computing- A Systematic Review. *International Journal of Computer Applications (0975 – 8887)*
8. Chen, S. L., Chen, Y. Y., & Kuo, S. H. (2017). CLB: A novel load balancing architecture and algorithm for cloud services. *Computers & Electrical Engineering*, *58*, 154-160.
9. Danlami Gabi, Abdul Samad Ismail& Anazida Zainal(2015). Systematic Review on Existing Load Balancing Techniques in Cloud Computing. *International Journal of Computer Applications (0975 – 8887)*
10. Chaudhury, K. S., Pattnaik, S., Moharana, H. S., & Pradhan, S. (2020). Static load balancing algorithms in cloud computing: challenges and solutions. In *International Conference on Soft Computing and Signal Processing* (pp. 259-265). Springer, Singapore.
11. Jiang, Y. (2015). A survey of task allocation and load balancing in distributed systems. *IEEE Transactions on Parallel and Distributed Systems*, *27*(2), 585-599.
12. Singh, S., & Chana, I. (2016). A survey on resource scheduling in cloud computing: Issues and challenges. *Journal of grid computing*, *14*(2), 217-264.
13. Milani, A. S., & Navimipour, N. J. (2016). Load balancing mechanisms and techniques in the cloud environments: Systematic literature review and future trends. *Journal of Network and Computer Applications*, *71*, 86-98.Hota, A., Mohapatra, S., & Mohanty, S. (2019). Survey of different load balancing approach-based algorithms in cloud computing: a comprehensive review. *Computational intelligence in data mining*, 99-110.
14. Xu, M., Tian, W., & Buyya, R. (2017). A survey on load balancing algorithms for virtual machines placement in cloud computing. *Concurrency and Computation: Practice and Experience*, *29*(12), e4123.
15. Thakur, A., & Goraya, M. S. (2017). A taxonomic survey on load balancing in cloud. *Journal of Network and Computer Applications*, *98*, 43-57.
16. Hota, A., Mohapatra, S., & Mohanty, S. (2019). Survey of different load balancing approach-based algorithms in cloud computing: a comprehensive review. *Computational intelligence in data mining*, 99-110.

17. Jyoti, A., Shrimali, M., Tiwari, S., & Singh, H. P. (2020). Cloud computing using load balancing and service broker policy for IT service: a taxonomy and survey. *Journal of Ambient Intelligence and Humanized Computing*, *11*(11), 4785-4814.
18. Elmagzoub, M. A., Syed, D., Shaikh, A., Islam, N., Alghamdi, A., & Rizwan, S. (2021). A survey of swarm intelligence based load balancing techniques in cloud computing environment. *Electronics*, *10*(21), 2718.
19. Balaji, K. (2021). Load balancing in cloud computing: issues and challenges. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(2), 3077-3084.
20. Ray, S., & De Sarkar, A. (2012). Execution analysis of load balancing algorithms in cloud computing environment. *International Journal on Cloud Computing: Services and Architecture (IJCCSA)*, *2*(5), 1-13.
21. Khan, R. Z., & Ahmad, M. O. (2016). Load balancing challenges in cloud computing: a survey. In *Proceedings of the International Conference on Signal, Networks, Computing, and Systems* (pp. 25-32). Springer, New Delhi.
22. Daraghmi, E. Y., & Yuan, S. M. (2015). A small world based overlay network for improving dynamic load-balancing. *Journal of Systems and Software*, *107*, 187-203.
23. Abdulhamid, S. I. M., Abd Latiff, M. S., & Bashir, M. B. (2014). Scheduling techniques in on-demand grid as a service cloud: a review.
24. Ramezani, F., Lu, J., & Hussain, F. K. (2014). Task-based system load balancing in cloud computing using particle swarm optimization. *International journal of parallel programming*, *42*(5), 739-754.
25. Rastogi, G., & Sushil, R. (2015, October). Analytical literature survey on existing load balancing schemes in cloud computing. In *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)* (pp. 1506-1510). IEEE.
26. Abdullahi, M., & Ngadi, M. A. (2016). Symbiotic organism search optimization based task scheduling in cloud computing environment. *Future Generation Computer Systems*, *56*, 640-650.
27. Nakai, A., Madeira, E., & Buzato, L. E. (2015). On the use of resource reservation for web services load balancing. *Journal of Network and Systems Management*, *23*(3), 502-538.
28. LD, D. B., & Krishna, P. V. (2013). Honey bee behavior inspired load balancing of tasks in cloud computing environments. *Applied soft computing*, *13*(5), 2292-2303.
29. Rastogi, G., & Sushil, R. (2015, October). Analytical literature survey on existing load balancing schemes in cloud computing. In *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)* (pp. 1506-1510). IEEE.
30. Mishra, S. K., Sahoo, B., & Parida, P. P. (2020). Load balancing in cloud computing: a big picture. *Journal of King Saud University-Computer and Information Sciences*, *32*(2), 149-158.
31. Shafiq, D. A., Jhanjhi, N. Z., & Abdullah, A. (2021). Load balancing techniques in cloud computing environment: A review. *Journal of King Saud University-Computer and Information Sciences*.
32. Singh, A., Juneja, D., & Malhotra, M. (2015). Autonomous agent based load balancing algorithm in cloud computing. *Procedia Computer Science*, *45*, 832-841.
33. 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.
34. Alam, M., & Khan, Z. A. (2017). Issues and challenges of load balancing algorithm in cloud computing environment. *Indian journal of science and Technology*, *10*(25), 1-12.
35. Kalra, M., & Singh, S. (2015). A review of metaheuristic scheduling techniques in cloud computing. *Egyptian informatics journal*, *16*(3), 275-295.
36. Nishant, K., Sharma, P., Krishna, V., Gupta, C., Singh, K. P., & Rastogi, R. (2012, March). Load balancing of nodes in cloud using ant colony optimization. In *2012 UKSim 14th international conference on computer modelling and simulation* (pp. 3-8). IEEE.

37. Domanal, S. G., & Reddy, G. R. M. (2013, October). Load balancing in cloud computingusing modified throttled algorithm. In *2013 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)* (pp. 1-5). IEEE.
38. Cosenza, B., Cordasco, G., De Chiara, R., & Scarano, V. (2011, February). Distributed load balancing for parallel agent-based simulations. In *2011 19th International Euromicro Conference on Parallel, Distributed and Network-Based Processing* (pp. 62-69). IEEE.
39. Kanakala, V. R., Reddy, V. K., & Karthik, K. (2015, March). Performance analysis of load balancing techniques in cloud computing environment. In *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)* (pp. 1-6). IEEE.
40. Alakeel, A. M. (2010). A guide to dynamic load balancing in distributed computer systems. *International journal of computer science and information security*, *10*(6), 153-160.
41. Yahaya, B., Latip, R., Othman, M., & Abdullah, A. (2011). Dynamic load balancing policy with communication and computation elements in grid computing with multi-agent system integration. *International Journal on New Computer Architectures and Their Applications "(IJNCAA)*, *1*(3), 780-788.
42. Mukhopadhyay, R., Ghosh, D., & Mukherjee, N. (2010, February). A study on the application of existing load balancing algorithms for large, dynamic, heterogeneous distributed systems. In *Proc. 9th WSEAS Int. Conf. Software Engineering, Parallel and Distributed Systems (SEPADS)* (pp. 238-243).
43. Ahmad, F., Chakradhar, S. T., Raghunathan, A., & Vijaykumar, T. N. (2012). Tarazu: optimizing mapreduce on heterogeneous clusters. *ACM SIGARCH Computer Architecture News*, *40*(1), 61-74.
44. Houle, M. E., Symvonis, A., & Wood, D. R. (2002). Dimension-exchange algorithms for load balancing on trees. In *Procs. of 9th Int. Colloquium on Structural Information and Communication Complexity*.
45. Kokilavani, T., & Amalarethinam, D. G. (2011). Load balanced min-min algorithm for static meta-task scheduling in grid computing. *International Journal of Computer Applications*, *20*(2), 43-49.
46. Nusrat Pasha, Amit Agarwal, and Ravi Rastogi. 2014. Round robin approach for VM load balancing algorithm in cloudcomputing environment. International Journal of Advanced Research in Computer Science and Software Engineering 4, 5(2014), 34–39.
47. Bala, A., & Chana, I. (2016). Prediction-based proactive load balancing approach through VM migration. *Engineering with Computers*, *32*(4), 581-592.
48. Ghoneem, M., & Kulkarni, L. (2017). An adaptive MapReduce scheduler for scalable heterogeneous systems. In *Proceedings of the International Conference on Data Engineering and Communication Technology* (pp. 603-611). Springer, Singapore.
49. De Falco, I., Laskowski, E., Olejnik, R., Scafuri, U., Tarantino, E., & Tudruj, M. (2015). Extremal optimization applied to load balancing in execution of distributed programs. *Applied Soft Computing*, *30*, 501-513.
50. Remesh Babu, K. R., & Samuel, P. (2016). Enhanced bee colony algorithm for efficient load balancing and scheduling in cloud. In *Innovations in bio-inspired computing and applications* (pp. 67-78). Springer, Cham.
51. Devi, D. C., & Uthariaraj, V. R. (2016). Load balancing in cloud computing environment using improved weighted round robin algorithm for nonpreemptive dependent tasks. *The scientific world journal*, *2016*.
52. Liu, Y., Zhang, C., Li, B., & Niu, J. (2017). DeMS: A hybrid scheme of task scheduling and load balancing in computing clusters. *Journal of Network and Computer Applications*, *83*, 213-220.
53. Naha, R. K., & Othman, M. (2016). Cost-aware service brokering and performance sentient load balancing algorithms in the cloud. *Journal of Network and Computer Applications*, *75*, 47-57.
54. Tasquier. 2015. Agent based load-balancer for multi-cloud environments. Columbia International Publication Journal ofCloud Computing Research 1, 1 (2015), 35–49.

55. Gutierrez-Garcia, J. O., & Ramirez-Nafarrate, A. (2015). Agent-based load balancing in cloud data centers. *Cluster Computing*, *18*(3), 1041-1062.
56. Keshvadi, S., & Faghih, B. (2016). A multi-agent based load balancing system in IaaS cloud environment. *International Robotics & Automation Journal*, *1*(1), 1-6.
57. Shen, H., Yu, L., Chen, L., & Li, Z. (2016, December). Goodbye to fixed bandwidth reservation: Job scheduling with elastic bandwidth reservation in clouds. In *2016 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)* (pp. 1-8). IEEE.
58. Xin, Y., Xie, Z. Q., & Yang, J. (2017). A load balance oriented cost efficient scheduling method for parallel tasks. *Journal of Network and Computer Applications*, *81*, 37-46.
59. Elmougy, S., Sarhan, S., & Joundy, M. (2017). A novel hybrid of Shortest job first and round Robin with dynamic variable quantum time task scheduling technique. *Journal of Cloud computing*, *6*(1), 1-12.
60. Zhao, J., Yang, K., Wei, X., Ding, Y., Hu, L., & Xu, G. (2015). A heuristic clustering-based task deployment approach for load balancing using Bayes theorem in cloud environment. *IEEE Transactions on Parallel and Distributed Systems*, *27*(2), 305-316.
61. D'Agostini, G. (1995). A multidimensional unfolding method based on Bayes' theorem. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, *362*(2-3), 487-498.
62. Pawlak, Z. (2000, October). Rough sets and decision algorithms. In *International Conference on Rough Sets and Current Trends in Computing* (pp. 30-45). Springer, Berlin, Heidelberg.
63. Hussein, A. A., Yaseen, E. T., & Rashid, A. N. (2023). Learnheuristics in routing and scheduling problems: A review. *Samarra Journal of Pure and Applied Science*, *5*(1), 60-90.
64. Turki, A. I., & Hasson, S. T. (2023). Study Estimating hourly traffic flow using Artificial Neural Network: A M25 motorway case. *Samarra Journal of Pure and Applied Science*, *5*(1), 47-59.
65. Kang, B., & Choo, H. (2016). A cluster-based decentralized job dispatching for the large-scale cloud. *EURASIP Journal on Wireless Communications and Networking*, *2016*(1), 1-8.
66. Han, Y., & Chronopoulos, A. T. (2017). Scalable loop self-scheduling schemes for large-scale clusters and cloud systems. *International Journal of Parallel Programming*, *45*(3), 595-611.

IRAQI Academic Scientific Journals
Samarra Journal of Pure and Applied Science
SJPAS
مجلة سامراء للعلوم الصرفة والتطبيقية
www.sjpas.com
p ISSN: 2663-7405
e ISSN: 2789-6838

تقنيات موازنة الحمل في الحوسبة السحابية: مراجعة

محمد خوام احمد [1,2]* ، صلاح عواد سلمان[2] ، عمر يونس عبد الحميد[3]

1- المعهد التقني بعقوبة، الجامعة التقنية الوسطى، العراق
2- كلية علوم الحاسوب وتكنولوجيا المعلومات، جامعة الانبار، العراق
3- كلية العلوم، جامعة كرميان، العراق

معلومات البحث:

تأريخ الاستلام: 2023/05/14
تاريخ التعديل: 2023/06/25
تأريخ القبـــول: 2023/06/30
تاريخ الـنشر: 2024/03/30

الكلمات المفتاحية:

الحوسبة السحابية، الة افتراضية، موازنة الحمل، جودة الخدمة

معلومات المؤلف

الايميل:
الموبايل:

الخلاصة:

توفر الحوسبة السحابية إمكانية وصول سهلة ومرنة للموارد الموجودة على الإنترنت. في هذه الحالة، يمكن للعملاء استخدام الموارد المتاحة حسب حاجتهم دون ترقية أجهزتهم الخاصة. وبالتالي، تعتبر موازنة التحميل واحدة من أكثر القضايا صعوبة المتعلقة بالحوسبة السحابية حيث يجب تشغيل مهام (عمليات) متعددة في وقت واحد على عناصر المعالجة. هناك خوارزميات مختلفة تستخدم لتخصيص المهمة على تلك العناصر. يمكن توزيع المهام وفقًا لمخططات مختلفة. تقترح بعض الخوارزميات تحديد أولويات المهام بينما يقوم البعض الآخر بتوزيع الرصيد وفقًا لطول المهمة. ومع ذلك، هناك عدد كبير من أساليب موازنة التحميل التي تعتمد على تقنيات الذكاء الاصطناعي. وعلى وجه التحديد، استخدام الخوارزميات الفوقية لتوزيع المهام على الأجهزة الافتراضية. الهدف من هذه الخوارزميات هو تعزيز إنتاجية النظام السحابي من خلال البحث عن التوزيع الأمثل لتلك المهام على الأجهزة الافتراضية. هناك عدد كبير من المنهجيات التي تستحق المراجعة والتحقيق من حيث كفاءتها وأدائها وإنتاجيتها. الهدف الرئيسي من هذا العمل هو تقديم ورقة مراجعة شاملة للأدبيات التي تناقش التقدم في هذا المجال على مر السنين. ويتم دراسة مزايا وعيوب تلك الأساليب من أجل تسليط الضوء على الثغرات ومحاولة اقتراح بعض الحلول في المستقبل. وبالإضافة إلى ذلك، يتم إجراء التصنيف على أساس معاملات احصائية. الى جانب ذلك، تم إجراء تحليل كامل للطرق المقارنة. كما تم تسليط الضوء على الجوانب المستقبلية لاستخدام خوارزميات موازنة التحميل المستخدمة حاليًا.