JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

A literature survey on Load Balancing Algorithm in Green Cloud Computing

Mr. B. Alen Gamber,
Research Scholar,
Department of Computer Science
(VISTAS)
Chennai, TamilNadu, India.

Dr. K. Abirami,
Assistant professor, School of computing sciences,
(VISTAS)

Dr. K. Dharmarajan
Professor, School of computing sciences
(VISTAS)

Abstract: Cloud computing is an innovative technology that has been utilized more and more in recent years. Cloud computing is the storage and access of data on servers hosted on the internet in contrast to the traditional method where the data will be stored on the local server or the hard drive in the system. More and more companies are adopting cloud computing in their company. Using these resources computing resources are provided to the user using the internet. As a result of this rapid expansion, the number of data centres, the palace where the data is stored, has tremendously increased which has resulted in an increase in power consumption which in turn raises sustainability concerns as the link between power consumption and carbon footprint is already recognized. Another challenge is that of load balancing which is the distribution of the workload among the available nodes to ensure that the load is spread evenly across the system and a single node does not have excess load. It means the efficient use of resources and reduces the use of unnecessary resources thereby saving energy. This energy-efficient cloud computing is denoted as the green cloud or green cloud computing and load balancing is one of the ways to achieve it. This paper attempts to identify and summarise the recent literature related to Load load-balancing algorithms in Green Cloud Computing and provides an overview of all the important concepts in this topic.

Keywords: cloud, cloud computing, load-balancing, green cloud computing, algorithm

1. Introduction

Cloud computing

Cloud computing is an innovative technology. It is a distributed computing paradigm that enables users to access virtual resources like storage, networks, computers, applications etc. One of the advantages of cloud is that these mentioned resources can be scaled up and scaled down based on the requirements easily (González-Martínez et al., 2015). These characteristics of the cloud make it cost-effective and easy to maintain. After its introduction, it has been widely studied and adopted in many fields. Due to the tremendous advantages that the cloud provides for its users and its cost-effectiveness, many businesses are gradually changing to a cloud-based process (Ali et al., 2015).

Google (no date) defines cloud computing as “Cloud computing is the on-demand availability of computing resources (such as storage and infrastructure), as services over the internet. It eliminates the need for individuals and businesses to self-manage physical resources themselves, and only pay for what they use.”

The term cloud generally refers to the internet or a network and suggests that the cloud is something that is remote. Cloud has various advantages over conventional systems. Many applications that are utilized in everyday life such as mail, CRM, online conferences, etc., are all based on clouds. Cloud computing offers various capabilities without having to physically buy the resources or download applications. It reduces the need for physical computing infrastructure and reduces the cost associated with it and the running, installation and maintenance costs. Using this approach the client can access the data anywhere and at any time (Malik et al., 2018; Puthal et al., 2015).

There are four types of cloud deployment models (Radu, 2017):

1. Community cloud: It is used by organizations that have the same concerns or goals.
2. Public cloud: Used by general people.
3. Private cloud: Used by a single private organization
4. Hybrid cloud: Combination of two or more above-mentioned cloud models.

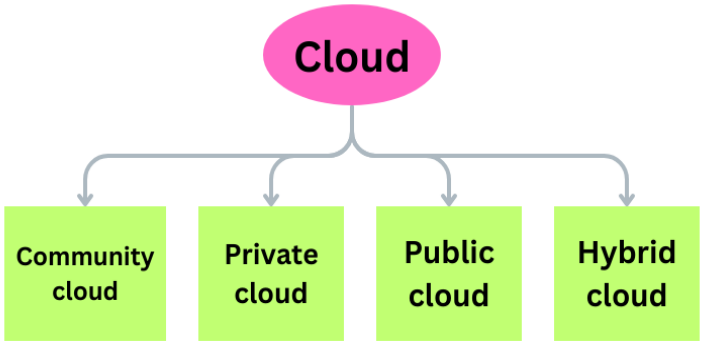

Figure 1: cloud deployment models

Some of the major characteristics of cloud computing are given below (Malik et al., 2018)

- Agility
- High scalability
- High availability and reliability
- Multi-sharing
- Rapid elasticity
- Low cost
- Maintenance
- On-Demand Self Service
- Services in pay-per-use mode
- Measured Service
- Broad network access
- Resource Pooling

Load Balancing in Cloud computing

Load balancing is a general term which denotes the allocation of a large processing load to a small processing node in order to improve the performance of the system. In a distributed system environment, it can be defined as the process of distribution of the load among the available nodes in the distributed system. This approach greatly improves job response time and resource utilization (Ala'Anzy and Othman, 2019; Fatima et al., 2019). For this purpose, various algorithms are present known as the load balancing algorithms. A good load-balancing algorithm must effectively distribute the workload without either underloading or overloading any node. However, for a cloud computing environment, selecting the load-balancing algorithm is a difficult task as various aspects like reliability, security, throughput, etc must be considered. Therefore, the major aim of a load-balancing algorithm in terms of cloud computing is to enhance the response time of the job by distributing the complete load of the system while ensuring that no other node is overloaded (Fatima et al., 2019; Devi and Uthariaraj, 2016).

Cloud computing is defined as a contemporary technology in the domain of computer science using which services can be provided to users at any time. In this technology, the cloud computing system is distributed all around the world in order to provide service to clients whenever required. Using any type of device such as phones, PDAs, laptops, etc., users can access data from anywhere anytime easily. Various challenges are associated with cloud computing and these include efficient load balancing, security, resource scheduling, data centre energy consumption, scaling, data lock-in and service availability, QoS management, and performance monitoring. One of the major challenges in cloud computing is described to be load-balancing where the loads must be assigned and reassigned to available resources. This load balancing results in enhanced throughput while decreasing the cost and response time, improving performance and resource utilization in addition to energy saving. With excellent load-balancing techniques, Service Level Agreement (SLA) and user satisfaction can also be provided. Therefore, providing efficient load-balancing algorithms and mechanisms is key to the success of cloud computing environments (Ghomi et al., 2017; Thakur and Goraya, 2017; Wen et al., 2015; Volkova et al., 2018; Bura et al., 2021; Deepa and Cheelu, 2017).

The process of load-balancing is generally automated to execute failover of the continuance of service on the failure of 1 or additional parts of it. The elements in cloud computing are all continuously monitored and if any element becomes unresponsive, the load balancer in that network is activated and prevents further data or traffic from reaching that element or node. When an appropriate load-balancing technique is used resource expenditure is significantly reduced. Apart from lowering the cost associated with cloud computing and making organizations go greener, the load-balancing technique reduces the stress on the circuits of the system which leads to a longer life of the system (Berwal and Kant, 2015; Gao and Wu, 2015; Samadi et al., 2018; Babu et al., 2015).

2. Literature Review

Load Balancing Algorithm in Green Cloud Computing

Out of 7.3 billion people in the world, nearly 55% of the population uses the internet. This creates a significant demand for information technology services for both organizations and consumers. Therefore, in order to meet this growing demand, the number of data centres is also increasing and is estimated to consume 2% of the global energy. Due to the ability of cloud computing to meet the needs of the user

when requested majority of the companies are adopting this technology. In the use of cloud computing and its organizational design, green cloud computing has become an important aspect to be taken into account. One way to effectively green the cloud is by efficiently using cloud-based resources. In this paper, the researcher specifically focuses on resource allocation of cloud computing technologies to green an organization's cloud. The researchers designed a pricing model and an allocation model focused on resources of private cloud computing. Using this methodology firms can effectively load balance their cloud computing resources. The researchers also designed a pricing model to be used along with the service. This dynamic pricing model. Maximizes the net value of users in a private cloud computing service. Additionally, a job allocation algorithm based on our dynamic pricing model to load balance the cloud is also created. It is demonstrated that balancing the job allocation such that the number of jobs at individual resource servers is as close to equal as possible is optimal. Furthermore, simulations were run on the allocation mechanism to examine the effects on the cloud resources while gaining insight into the effective distribution of resources to public clouds. The model proposed by the researchers can help organizations to efficiently distribute their cloud-based resources, which allows for a greener cloud computing system (Kumar et al., 2022).

A green cloud is defined as the reduction of energy consumption by the data centres. This helps in reducing the waste disposed into the environment. Due to the increase in the demand for cloud computing, the energy consumption associated with it is also increasing. One way to achieve a green cloud is by server consolidation and proper load balancing techniques using virtual machine (VM) migration which is a feature provided by virtualization. The virtual machines are transferred from one host to another. The overhead associated with migration is performance degradation. This can be overcome by proper load-balancing techniques. This helps to curtail the number of VM migrations as well as energy consumption. The researchers in this paper propose a technique called threshold compare and load balance algorithm (TCLBA) which can be used to optimize resources on the provider's side. Two thresholds, upper and lower are defined for the purpose of load balancing. The algorithm works based on these threshold levels and shifts the load from a server if the load is above the upper threshold or to the server if its load is below the lower threshold. The load is balanced by migrating the VMs. The workload is consolidated to a smaller number of hosts such that the remaining hosts are shut down. This method solves the purpose of effective utilization of available resources with lesser energy consumption (Shahapure et al., 2021)

Hasan et al., (2018) state that the quality of the server used in cloud computing might be affected when the resources are limited which leads to poor service delivery to the consumers. This suggests a need for an efficient load-balancing technique in order to optimize the resources. Cloud servers with correlated load balancers can assist in optimizing the load balancing practicability in cloud computing. The researchers have proposed a load-balancing model (DLBS) for load balancing and effectively optimizing resources. This proposed methodology was designed by employing the Amazon EC2. Every server is equipped with a load balancer which observes the load and sends status information to the controller. The servers with fewer loads were given more requests while the overloaded ones were not given further requests. The Amazon Web Services (AWS) was used to demonstrate the proof of concept. The results revealed that the proposed solution improved performance, throughput, and utilization of cloud resources.

Khedr et al., (2015) highlight that the demand for cloud computing is rising which has increased the energy consumption of data centres, and has now become a critical issue. The researchers also mention that the data centers that host these cloud computing infrastructures consume a tremendous amount of power which in turn increases the cost as well as the carbon footprint associated with the process which is harmful to the environment. So, the researchers stress that a new green cloud load balancing (GCLB) solution that is intended to reduce energy consumption in cloud data centres while maintaining the service level agreement (SLA) between the customer and the cloud service Provider must be proposed. Therefore to design such solutions, deep analysis of the Cloud is required concerning their power efficiency. The researchers further discuss various elements of green clouds which contribute to the total energy consumption and how it is addressed in the previous studies. They also discuss the implication of this solution on energy efficiency (EE) and the quality of service (QOS).

Scheduling or the allocation of tasks in the cloud environment comes under the NP-hard optimization problem. In accordance with the cloud infrastructure and the user requests, the cloud system is assigned some load (that may be underloaded or overloaded or the load is balanced). Circumstances like underloaded and overloaded cause different system failures concerning power consumption, execution time, machine failure, etc. So, load balancing is needed to face all the above-mentioned problems. This load balancing of tasks on virtual machines (VMs) is a significant aspect of task scheduling in clouds. There are various types of loads in the cloud network such as memory load, Computation (CPU) load, network load, etc. Load balancing is the mechanism of detecting overloaded and underloaded nodes and then balancing the load among them. Therefore with the above information in mind, the researchers put forward load-balancing approaches in cloud computing to optimize different performance parameters. For the load-balancing algorithms, a taxonomy was also presented. A brief explanation of considered performance parameters in the literature and their effects is presented in this paper. To analyze the performance of heuristic-based algorithms, the simulation is carried out in the CloudSim simulator and the results are presented in detail (Mishra et al., 2020).

Geetha and Rene Robin (2021) highlight that the goal of green cloud computing is to deal with the force and vitality effectiveness, the decision of eco-neighbourly equipment and programming, and reusing the material to build the item's life. Load balancing needs to consider the heterogeneous sort of assets in the cloud server farm alongside its present state while choosing the allotment of client assignments to the asset. Cloud computing spins around Web-based securing and the arrival of assets from a data centre. Distributed computing like Web-based dynamic processing will likely experience the negative effects of over-burdening of solicitations. Other difficulties the system faces are asset usage in a cloud server farm and the nature of administration to the end clients because of inappropriate outstanding task-at-hand balances among accessible assets. Therefore, the researchers propose a new approach of time and energy-efficient load balancing (TELB) is introduced. This algorithm is based on time parameters and uses effective load balancing and scheduling of resources to consume more energy as well as within a minimum duration while executing the millions of tasks from various regions. This proposed algorithm has been actualized and found to give good outcomes of accurate, predictable, and reliable.

Cloud computing dynamically allocates virtual resources as per the demands of users. The rapid increase of data computation and storage in a cloud computing environment results in uneven distribution of workload on its heterogeneous resources. As a result of that, overloaded servers will have a higher job completion time compared to the corresponding time taken by underloaded servers in the same environment. Distributing a balanced workload over the available resources is a key challenge in the cloud computing environment. Generally, load

balancing is used to distribute the workload among multiple servers and to avoid overloading and underloading of servers. It helps to enhance the performance of the system and utilize the system more fairly. Therefore the researchers have put forward a novel hybrid load balancing approach in a cloud computing environment using Grey Wolf Optimization based Particle Swarm Optimization and compare it with Harmony Search, Particle Swarm Optimization Artificial Bee Colony, and Grey Wolf Optimization algorithms. It also helps to improve system performance and fair utilization of resources. The results of the research experiments were determined to be significant with improved convergence and simplicity (Gohil and Patel, 2018).

The researchers mention that the existing Load Balancing techniques mainly focus on reducing overhead, service response time improving performance etc., but none of the techniques has considered the energy consumption and carbon emission factors. They have proposed a Type of two-level centralized scheduling model along with a Global Centralized Scheduler (GCS) at a higher level and a Local Centralized Scheduler (LCS) at the next level to overcome the high communication cost of distributed algorithms and the single-point-of-failure problem of centralized algorithms. A load-balancing technique which is energy efficient can be utilized to enhance the cloud computing performance by maximum resource utilization and balancing the workload across the cloud nodes. This consequently reduces power consumption and carbon emissions significantly. In this approach as soon as the data center receives the new request for service it queries the global centralized scheduler for the allocation of a virtual machine. The global centralised scheduler gathers the details about the load from all the local centralised schedulers and will be redirected to the appropriate local centralised schedulers. The global centralised scheduler assumes the responsibility provided by the consumer and returns the results of the scheduling to the user. The local centralised scheduler gathers the load information from the computing nodes in that area and balances the load in that local area. Various tasks are assigned to diverse computing nodes when the local centralized schedulers receive requests from the global centralized scheduler. The local centralized schedulers also assemble the results from each and every computing node and then transmit them to the global centralized scheduler (Megharaj and Mohan, 2013)

Panwar and Mallick (2015) highlight that in the current scenario, Cloud computing has become a crucial jargon in the world of Technology and is the next stage in the evolution of the Internet. The issue of load balancing in cloud computing is crucial for the seamless operation of cloud computing systems and plays a significant role in preventing challenges to the rapid advancement of cloud computing. In the modern era, a global demand for high-speed services is evident. Despite the existence of several load-balancing algorithms that efficiently allocate requests by selecting appropriate virtual machines, the challenge remains pertinent. The current paper introduces a dynamic load management algorithm designed to efficiently distribute incoming requests among virtual machines. Furthermore, the performance is simulated using the Cloud Analyst simulator, considering parameters such as data processing time and response time. A comparative analysis is conducted with the previously developed VM-Assign algorithm. The simulation results indicate that the proposed algorithm uniformly distributes the load among servers, effectively utilizing resources.

The issue of load balancing is described to be a multi-constraint and multi-variant problem that lowers the performance and the efficiency of the computing resources. Load balancing techniques provide a solution for load unbalancing situations for two undesirable aspects namely overloading and under-loading. Despite the crucial significance of load-balancing techniques, there is currently a lack of a comprehensive, systematic, extensive, and hierarchical classification for existing load-balancing techniques. Additionally, the literature has not delved into the study or consideration of factors that contribute to load unbalancing problems. This paper provides an exhaustive and comprehensive review of load-balancing techniques. It emphasizes the advantages and limitations of current methods, addressing critical challenges to pave the way for the development of more efficient load-balancing algorithms in the subsequent years. Additionally, the paper puts forward novel insights for the improvement of load balancing in cloud computing (Afzal and Kavitha, 2019).

The researchers emphasize the growing popularity of cloud computing, attributed to its appealing features, leading to a substantial increase in the load on cloud resources. Load balancing plays a crucial role in the cloud computing environment by ensuring equitable distribution of workload among devices or processors within the same timeframe. Various models and algorithms have been developed for load balancing in cloud computing, aiming to provide end users with easy and convenient access to cloud resources. The researchers aim to provide a systematic and thorough examination of research on load-balancing algorithms in cloud computing. This paper conducts a survey of state-of-the-art load-balancing tools and techniques from the years 2004 to 2015. The existing approaches, geared towards achieving equitable load balancing, are categorized to offer a clear and comprehensive understanding. This classification allows for an easy and concise overview of the underlying models adopted by each approach (Aslam and Shah, 2015).

Cloud Computing stands out as an emerging domain in the realm of Information Technology (IT), characterized by its internet-based nature, utility emphasis, and adherence to the pay-as-you-go model. Within this context, load balancing emerges as a crucial concern in cloud computing. This technique leverages multiple nodes to distribute dynamic workloads among them, ensuring that no single node becomes overloaded. The primary objectives of load balancing encompass optimal resource utilization to enhance system performance and the minimization of resource consumption to reduce carbon emission rates. The primary focus of this paper is load-balancing techniques in cloud computing. The review is instrumental in analyzing the challenges posed by current load-balancing algorithms and provides a comparative assessment based on various qualitative metrics such as throughput, reliability, power-saving features, performance, scalability, and associated overhead (Mishra and Mishra, 2015).

Cloud computing technology uses load balancing and scheduling for virtualized file sharing in cloud infrastructure. The above-mentioned functions have to be performed in an optimised manner in order to achieve optimal file sharing. In the domain of cloud data centres, scalable traffic management has emerged to address traffic load balancing and quality of service provisioning. Despite advancements, reducing latency in multidimensional resource allocation remains a challenge. Therefore, there is a pressing need for an effective resource scheduling approach to optimize load in the cloud. This study aims to present an integrated algorithm for resource scheduling and load balancing to enhance cloud service provisioning. The proposed method involves the creation of a Fuzzy-based Multidimensional Resource Scheduling model to achieve efficiency in resource scheduling within the cloud infrastructure. With the aid of fair and effective load-balancing utilization of Virtual Machines can be enhanced by dynamically selecting a request from a class using the Multidimensional Queuing Load Optimization algorithm. Next, to avoid underutilization and overutilization of resources A load balancing algorithm is then implemented. improving latency time for

each class of request. Simulations were conducted to evaluate the effectiveness of using the Cloudsim simulator in cloud data centres and the consequences show that the suggested method accomplishes improved performance in terms of average resource scheduling efficiency, success rate, and response time. Simulation analysis shows that the method improves the resource scheduling efficiency by 7% and also reduces the response time by 35.5 % when compared to state-of-the-art works (Priya et al., 2019).

The domain of cloud computing is a technology that is growing rapidly. It is being implemented in a variety of domains like research, industry, business, and computing. Cloud computing offers diverse services over the internet, eliminating the necessity for personalized hardware and other dedicated resources. However, cloud computing environments encounter challenges related to resource utilization, energy efficiency, and the heterogeneity of resources. To address these issues, task scheduling and virtual machines (VMs) serve as consolidation techniques. Extensive research has been conducted on task scheduling in the literature to mitigate these challenges. Various studies have investigated the problem with diverse parameters and objectives. This paper specifically focuses on the challenge of energy consumption and optimizing resource utilization within virtualized cloud data centres. The proposed algorithm centres around task classification and the implementation of thresholds to enhance scheduling efficiency and overall resource utilization. In the initial phase, workflow tasks undergo preprocessing to mitigate bottlenecks, achieved by segregating tasks with numerous dependencies and extended execution times into separate queues. Next, the classification of the tasks is carried out on the basis of required resource intensities. As the final step, Particle Swarm Optimization (PSO) is used to select the best schedules. Experiments were performed to validate the proposed technique. The researchers have presented their outcomes of the comparative results which illustrate the effectiveness of the proposed algorithm over that of the other algorithms to which it was compared in terms of energy consumption, makespan, and load balancing (Malik et al., 2021).

Efficient task load balancing in the cloud environment plays a crucial role in distributing resources from data centers. The dynamic nature of computing through the internet results in cloud computing facing challenges such as overloading of requests. Load balancing becomes imperative, and it needs to be executed in a way that ensures all virtual machines (VM) maintain a balanced load for optimal utilization of their capabilities. This paper introduces a novel approach for dynamically balancing load among virtual machines by employing a hybridization of a modified Particle Swarm Optimization (MPSO) and an enhanced Q-learning algorithm known as QMPSO. The hybridization process involves adjusting the velocity of the modified Particle Swarm Optimization (MPSO) based on the best action generated through the improved Q-learning. This hybrid approach aims to improve machine performance by achieving load balancing among virtual machines (VMs), maximizing VM throughput, and optimizing task waiting times to maintain a balance between task priorities. The algorithm's robustness is validated through a comparison of results from the simulation process with existing load balancing and scheduling algorithms. The comparison, conducted on both simulation and real platforms, demonstrates the superior performance of our proposed algorithm over its competitors (Jena et al., 2022).

A novel perspective for provisioning large-scale computing resources by using virtualization technology and a pay-per-use cost model has been offered by Cloud computing. Load balancing is considered to be a crucial part of distributed and parallel systems. This aids cloud computing to enhance performance, better utilization of computing resources, management of energy consumption enhancing the cloud services' QoS, avoiding SLA violations and maintaining system stability through distribution, controlling and managing the system workloads. This paper explores the essential requirements and considerations involved in designing and implementing an effective load balancer for cloud environments. Additionally, it presents a comprehensive survey of currently proposed cloud load-balancing solutions, categorized into three main types: General Algorithm-based, Architectural-based, and Artificial Intelligence-based load-balancing mechanisms. The proposed model evaluates these solutions using relevant metrics and provides a thorough discussion of their respective advantages and disadvantages (Mesbahi and Rahmani, 2016).

In the context of a cloud data centre, the primary challenge lies in addressing the dynamic stream of billions of requests from end-users. Efficiently managing and handling such requests requires the equitable distribution of the load among the cloud nodes. To attain this objective, numerous load-balancing approaches have been put forth in recent years. The strategies employed in load balancing aim to enhance user satisfaction by minimizing task response times and optimizing resource utilization through the fair and even allocation of cloud resources. The conventional Throttled load balancing algorithm is a good approach for load balancing in cloud computing as it distributes the incoming jobs evenly among the VMs. However, the important drawback was that the algorithm worked effectively in homogeneous VMS environments but it did not take into account resource-specific demands of the tasks and had the additional overhead of scanning the entire list of VMs every time a task came. The researchers in this paper have addressed the above-mentioned problems by proposing an algorithm for Cluster-based load balancing that operates properly in heterogeneous node environments, estimates resource-specific demands of the tasks and minimizes scanning overhead by splitting the machines into clusters. Experimental consequences have demonstrated that the proposed algorithm gives better results in terms of execution time, waiting time, turnaround time and throughput as compared to existing throttled and modified throttled algorithms (Kapoor and Dabas, 2015).

In spite of the conducting various resources in the field of cloud computing serval issues still exist in the load balancing especially in the Infrastructure as service (IaaS) cloud model. The vital step in cloud computing is the allocation of tasks as a result of restricted virtual machines or resources. The service providers should make sure that high service delivery performance in similar models, preventing circumstances such as hosts being overloaded or underloaded as it might result in greater execution time or machine failure, etc. Task scheduling plays a pivotal role in load balancing, with task scheduling aligning closely with the requirement outlined in the Service Level Agreement (SLA), a document provided by cloud developers to users. Key SLA parameters, including deadlines, are taken into consideration in the Load Balancing (LB) algorithm. The primary objective of the proposed algorithm is to optimize resources and enhance load balancing, particularly in light of the Quality of Service (QoS) task parameters, prioritization of virtual machines (VMs), and resource allocation. This proposed algorithm solves the highlighted problems as well as the existing research gap in the findings of the review of the literature. The outcomes indicate that the introduced Load Balancing (LB) algorithm leads to an average resource utilization of 78%, surpassing the existing Dynamic LBA algorithm. Additionally, the proposed algorithm demonstrates commendable performance, showcasing reduced execution time and makespan (Shafiq et al., 2021).

The execution of the cloud infrastructure is heavily reliant on task scheduling and load balancing. Consequently, numerous load-balancing algorithms and techniques have been proposed by researchers globally. These approaches share the common goal of fairly distributing the workload among all virtual machines to achieve optimal efficiency. In this research paper, the researchers developed an algorithm for load balancing which aims to reduce the makeson time and increase the utilization ratio of cloud resources. The results of the computation reveal that the development of an algorithm declines the makespan time and improves the utilization of resources compared to the fcfs, min-min algorithm and shortest job first in all conditions. The presented methodology is a dynamic load-balancing algorithm designed to minimize the makespan time and enhance the average resource utilization ratio within a cloud environment. In the simulation process, the expected processing time for each task on the virtual machine was initially determined. Subsequently, the count of overloaded, underloaded, and balanced virtual machines was ascertained. Various algorithms exist in the cloud environment, including heuristic-based algorithms, metaheuristic-based algorithms, and conventional approach-based algorithms. Each algorithm operates on distinct parameters such as makespan time, execution time, response time, resource utilization, throughput, etc., optimizing these parameters on the basis of an objective function. The conventional proposed methodology aims to fairly balance the load in the cloud by utilizing a task migration approach. The proposed algorithm not just reduces the makespan duration but additionally decreases the possibility of overloading and underloading a virtual machine. From the results of the experiment, it was revealed that under all feasible conditions suggested algorithm lowers the makespan time and raises the average resource utilization ratio compared with SJF, FCFS and Min-Min (Kumar and Sharma, 2017).

The objective of virtual machine (VM) scheduling with load balancing in cloud computing is to allocate VMs to appropriate servers and maintain a balanced distribution of resources across all servers. In the context of an infrastructure-as-a-service framework, the system deals with dynamic input requests, involving the creation of VMs without prior knowledge of the types of tasks to be executed on them. Consequently, scheduling methods that concentrate solely on fixed task sets or demand detailed task information are not well-suited for this system. This reserach binds ant colony optimization and particle swarm optimization to resolve the VM scheduling complication, with the result being known as ant colony optimization with particle swarm (ACOPS). It employs historical data in order to estimate the workload of the new input requests to adapt to the environment which is dynamic in nature excluding additional task information. ACOPS also deny requests that are unable to be satisfied before scheduling to mitigate the computing time of the scheduling technique. Experimental outcomes suggest that the submitted algorithm can keep the load balance in a dynamic environment and surpass other approaches (Cho et al., 2015).

As the field of cloud computing rapidly evolves, effective management of resource allocation processes becomes imperative. This paper introduces a load-balancing algorithm based on honey bee behaviour (LBA_HB). The primary objective is to distribute the workload across multiple network links in a manner that mitigates both underutilization and overutilization of resources. This is achieved by assigning incoming tasks to virtual machines (VMs) that satisfy two conditions: firstly, the number of tasks currently processed by the chosen VM is fewer than those processed by other VMs, and secondly, the deviation of the processing time for this VM from the average processing time of all VMs is below a specified threshold value. The suggested algorithm is analyzed in comparison with various scheduling algorithms ant colony, honey bee, modified throttled and round robin algorithms. The experimental results demonstrate the effectiveness of the proposed algorithm across various metrics, including execution time, response time, makespan, standard deviation of load, and degree of imbalance. Rooted in the natural foraging behaviour of honey bees, the algorithm mirrors the efficiency of these insects. The assigned task communicates the status of the virtual machine (VM) to the remaining tasks, akin to bees sharing information about an abundant food source through their waggle dance in the hive. The LBA_HB algorithm is implemented and simulated using CloudSim. Comparative analysis involves benchmarking against both conventional and SI-based load balancing algorithms, including round robin, modified throttled, ant colony, and honey bee algorithms. The outcomes of the experiments illustrate the effectiveness of LBA_HB concerning response time, standard deviation of load, makespan, and degree of imbalance. The primary objective of the proposed Load-Balancing Algorithm based on Honey Bee behavior (LBA_HB) is to distribute workload effectively, mitigating both underutilization and overutilization of resources. This is achieved by assigning incoming tasks to a virtual machine (VM) that satisfies two conditions: firstly, the number of tasks currently processed by this VM is fewer than those processed by other VMs, and secondly, the deviation of this VM's processing time from the average processing time of all VMs is below a predefined threshold value (Hashem et al., 2017).

Though cloud computing is a rapidly growing and adapted technology in various domains, there are still some drawbacks to its application like that of load balancing. Load balancing is a technique that dynamically distributes workloads among various nodes, ensuring a fair distribution, particularly in scenarios where some nodes are underloaded while others are overloaded. The key accomplishments of load balancing include optimizing resource consumption and reducing energy consumption. Swarm intelligence plays a crucial role in addressing problems that are challenging to solve using classical and mathematical techniques, offering valuable insights and approaches in such scenarios. An artificial bee colony is foraging behaviour and the algorithm is inspired by it. It was created in 2005 by Karaboga. This algorithm possesses fast convergence, powerful, robustness, and has high flexibility. Load balancing has been used by various researchers in order to enhance the process of load balancing. In this paper, the researchers have carried out a comprehensive study of load balancing in cloud computing using the ABC algorithm. Additionally, they also provide an overview of the basic concepts related to swarm intelligence and its qualities. Karaboga formulated the ABC algorithm based on the foraging characteristics of honey bee swarm intelligence. Meta-heuristic is defined to be a Greek word and it comprises Meta which means high-level and heuristics means to find or to know. Meta-heuristic is a group of intelligent steps which increase the effectiveness of a heuristic procedure. An ABC algorithm is a nature-enthused that is established on the foraging behaviour of bees. ABC algorithm optimization is excellent at consideration but poor at manipulation. The artificial algorithm is suggested for optimization techniques and they used intelligent foraging behaviour of honey bees (Ullah et al., 2019).

As mentioned above load balancing is a vital task in cloud computing. Cloud servers require to storage of an immense amount of data which boosts the load on the servers. Therefore the load-balancing technique is utilised to distribute the load equally with less energy consumption. In accordance with the abovementioned information, this research paper put forward a load-balancing technique based on the constraint measure. At the beginning, the capacity and load of each virtual machine are measured. In case if the load of the VM is more than the balanced threshold value subsequently, the load balancing algorithm is used for allocating the tasks. The load balancing algorithm computes the determinant factor of each VM and checks the load. Afterwards, it computes the selection factor of each task. Then, the task which has a superior selection factor is assigned to the virtual machine. The performance of the suggested load balancing procedure is assessed with the existent load balancing methods, such as HBB-LB, DLB, and HDLB for the evaluation metrics load and capacity. The

experiment demonstrates that the suggested method migrates only three tasks while the existing method HDLB migrates seven tasks (Polepally and Shahu Chatrapati, 2019).

In the domain of cloud computing (CC), task load balancing continues to be an essential problem of resource sharing from a data centre to guarantee that each VM has a balanced load to achieve maximum utilization of its capabilities. Load balancing is defined to be a NonPolynomial (NP) problem resolved with metaheuristic algorithms. A novel Quasi Oppositional Dragonfly Algorithm for Load Balancing (QODA-LB) was advanced to accomplish optimal resource scheduling in a CC environment. The suggested QODA-LB algorithm utilizes three variables to calculate an objective function: running cost, run time, and load. The QODA-LB algorithm assigns tasks to the VM based on its potential and the derivative objective function. Moreover, the proposed algorithm utilizes the Quasi-Oppositional Based Learning (QOBL) principle to enhance the convergence rate of the standard Dragonfly Algorithm (DA). An extensive series of tests were carried out, and the outcomes were examined in diverse ways to secure the efficient execution enhanced by the QODA-LB algorithm. The results of the simulation showed optimal load-balancing efficiency and exceeded the foremost approaches (Latchoumi and Parthiban, 2022).

Cloud computing is described as evolving as a novel model of big-scale distributed computing. It offers the required services based on an online on-demand and pay-as-you-go basis. In the CC environment, load balancing is a vital factor to be concentrated on which requires the distribution of the dynamic workload over numerous machines to make certain that no single machine is overloaded. This helps the system to utilize only the ideal amount of machines which results in an enhanced system performance. For this purpose, an efficient task-scheduling algorithm is needed. One of the simplest algorithms is the MinMin algorithm which provides a schedule that decreases the makespan but this algorithm fails to take advantage of resources effectively. In this paper, the researchers proposed an Improved load-balanced Min-Min (ILBMM) algorithm through a genetic algorithm (GA) in order to reduce the makespan and enhance the utilization of resources. The implementation of the algorithm has been accomplished using the CloudSim simulator and simulation consequences display that the proposed algorithm exceeds to current algorithm (Rajput and Kushwah, 2016).

3. Conclusion

Cloud computing is one of the advanced and innovative technologies that is being implemented in day-to-day life increasingly by both individuals and organizations. The aim of the paper is to conduct a literature survey on the latest research conducted on the topic of load balancing for green cloud computing. As stated above the usage of cloud computing is growing tremendously and so is the increase of the data centres where the cloud stores the data. This has resulted in a huge increase in energy consumption and its associated carbon footprint. Therefore to reduce this problem load balancing is utilized. The function fo load-balancing is to evenly distribute the workload to the servers so they do not get overloaded and also ensure that they are not utilized resulting in an efficient use of resources and a decrease in power consumption thereby reducing the carbon footprint associated with the process. This reduction of carbon footprint and ideal resource usage is termed as the green cloud or green cloud computing. There are various algorithms used for load balancing. The various techniques of load balancing are presented in this paper along with an overview of the vital concepts of cloud computing.

References

1. Afzal, S., & Kavitha, G. (2019). Load balancing in cloud computing–A hierarchical taxonomical classification. Journal of Cloud Computing, 8(1), 22.
2. Ala'Anzy, M., & Othman, M. (2019). Load balancing and server consolidation in cloud computing environments: a meta-study. IEEE Access, 7, 141868-141887.
3. Ali, M., Khan, S. U., & Vasilakos, A. V. (2015). Security in cloud computing: Opportunities and challenges. Information sciences, 305, 357-383.
4. Alyouzbaki, Y. A. G., & Al-Rawi, M. F. (2021). Novel load balancing approach based on ant colony optimization technique in cloud computing. Bulletin of Electrical Engineering and Informatics, 10(4), 2320-2326.
5. Aslam, S., & Shah, M. A. (2015, December). Load balancing algorithms in cloud computing: A survey of modern techniques. In 2015 National software engineering conference (NSEC) (pp. 30-35). IEEE.
6. Babu, K. R., Joy, A. A., & Samuel, P. (2015, September). Load balancing of tasks in cloud computing environment based on bee colony algorithm. In 2015 Fifth International Conference on Advances in Computing and Communications (ICACC) (pp. 89-93). IEEE.
7. Badotra, S., & Singh, J. (2019). INTRODUCTION TO LOAD BALANCING AND STRATEGIES USED IN SOFTWARE DEFINED NETWORKING. Innovations.
8. Berwal, M., & Kant, C. (2015). Load Balancing in cloud computing. Int. J. Comput. Sci. Commun, 6, 52-58.
9. Bura, D., Singh, M., & Nandal, P. (2021). Analysis and development of load balancing algorithms in cloud computing. In Research Anthology on Architectures, Frameworks, and Integration Strategies for Distributed and Cloud Computing (pp. 1177-1197). IGI Global.
10. Cho, K. M., Tsai, P. W., Tsai, C. W., & Yang, C. S. (2015). A hybrid meta-heuristic algorithm for VM scheduling with load balancing in cloud computing. Neural Computing and Applications, 26, 1297-1309.
11. Deepa, T., & Cheelu, D. (2017, August). A comparative study of static and dynamic load balancing algorithms in cloud computing. In 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS) (pp. 3375-3378). IEEE.
12. 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.
13. Gao, R., & Wu, J. (2015). Dynamic load balancing strategy for cloud computing with ant colony optimization. Future Internet, 7(4), 465-483.
14. Gauhar Fatima, S., Kausar Fatima, S., Abdul Sattar, S., Ahmed Khan, N., & Adil, S. (2019). Cloud computing and load balancing. International Journal of Advanced Research in Engineering and Technology, 10(2), 189-209.

15. Geetha, P., & Rene Robin, C. R. (2021). Time and Energy-Efficient Load Balancing Algorithm Toward Green Cloud Computing. In Inventive Communication and Computational Technologies: Proceedings of ICICCT 2020 (pp. 371-386). Springer Singapore.
16. 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.
17. Gohil, B. N., & Patel, D. R. (2018, August). A hybrid GWO-PSO algorithm for load balancing in cloud computing environment. In 2018 Second International Conference on Green Computing and Internet of Things (ICGCIoT) (pp. 185-191). IEEE.
18. González-Martínez, J. A., Bote-Lorenzo, M. L., Gómez-Sánchez, E., & Cano-Parra, R. (2015). Cloud computing and education: A state-of-the-art survey. Computers & Education, 80, 132-151.
19. Google (no date). What is Cloud Computing?. Google Cloud. Available at: https://cloud.google.com/learn/what-is-cloud-computing
20. Hasan, R. A., Mohammed, M. N., Ameedeen, M. A. B., & Khalaf, E. T. (2018). Dynamic load balancing model based on server status (DLBS) for green computing. Advanced Science Letters, 24(10), 7777-7782.
21. Hashem, W., Nashaat, H., & Rizk, R. (2017). Honey bee based load balancing in cloud computing. KSII Transactions on Internet & Information Systems, 11(12).
22. Jena, U. K., Das, P. K., & Kabat, M. R. (2022). Hybridization of meta-heuristic algorithm for load balancing in cloud computing environment. Journal of King Saud University-Computer and Information Sciences, 34(6), 2332-2342.
23. Kapoor, S., & Dabas, C. (2015, August). Cluster based load balancing in cloud computing. In 2015 Eighth International Conference on Contemporary Computing (IC3) (pp. 76-81). IEEE.
24. Khedr, A. E., Nasr, M., & Elmasry, H. (2015). New balancing technique for green cloud computing and environmental Sustainability. International Journal of Advanced Research, 3(9), 201-215.
25. Kumar, C., Marston, S., Sen, R., & Narisetty, A. (2022). Greening the cloud: a load balancing mechanism to optimize cloud computing networks. Journal of Management Information Systems, 39(2), 513-541.
26. Kumar, M., & Sharma, S. C. (2017). Dynamic load balancing algorithm for balancing the workload among virtual machine in cloud computing. Procedia computer science, 115, 322-329.
27. Latchoumi, T. P., & Parthiban, L. (2022). Quasi-oppositional dragonfly algorithm for load balancing in cloud computing environment. Wireless Personal Communications, 122(3), 2639-2656.
28. Malik, M. I., Wani, S. H., & Rashid, A. (2018). CLOUD COMPUTING-TECHNOLOGIES. International Journal of Advanced Research in Computer Science, 9(2).
29. Malik, N., Sardaraz, M., Tahir, M., Shah, B., Ali, G., & Moreira, F. (2021). Energy-efficient load balancing algorithm for workflow scheduling in cloud data centers using queuing and thresholds. Applied Sciences, 11(13), 5849.
30. Megharaj, G. C., & Mohan, K. G. (2013). Two level hierarchical model of load balancing in cloud. International Journal of Emerging Technology and Advanced Engineering, 3(10), 307-311.
31. Mesbahi, M., & Rahmani, A. M. (2016). Load balancing in cloud computing: a state of the art survey. Int. J. Mod. Educ. Comput. Sci, 8(3), 64.
32. Mishra, N. K., & Mishra, N. (2015). Load balancing techniques: need, objectives and major challenges in cloud computing-a systematic review. International Journal of Computer Applications, 131(18), 0975-8887.
33. 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.
34. Panwar, R., & Mallick, B. (2015, October). Load balancing in cloud computing using dynamic load management algorithm. In 2015 International Conference on Green Computing and Internet of Things (ICGCIoT) (pp. 773-778). IEEE.
35. Polepally, V., & Shahu Chatrapati, K. (2019). Dragonfly optimization and constraint measure-based load balancing in cloud computing. Cluster Computing, 22(Suppl 1), 1099-1111.
36. Priya, V., Kumar, C. S., & Kannan, R. (2019). Resource scheduling algorithm with load balancing for cloud service provisioning. Applied Soft Computing, 76, 416-424.
37. Puthal, D., Sahoo, B. P., Mishra, S., & Swain, S. (2015, January). Cloud computing features, issues, and challenges: a big picture. In 2015 International conference on computational intelligence and networks (pp. 116-123). IEEE.
38. Radu, L. D. (2017). Green cloud computing: A literature survey. Symmetry, 9(12), 295.
39. Rajput, S. S., & Kushwah, V. S. (2016, December). A genetic based improved load balanced min-min task scheduling algorithm for load balancing in cloud computing. In 2016 8th international conference on Computational Intelligence and Communication Networks (CICN) (pp. 677-681). IEEE.
40. Samadi, Y., Zbakh, M., & Tadonki, C. (2018, July). E-HEFT: enhancement heterogeneous earliest finish time algorithm for task scheduling based on load balancing in cloud computing. In 2018 International Conference on High Performance Computing & Simulation (HPCS) (pp. 601-609). IEEE.
41. Shafiq, D. A., Jhanjhi, N. Z., Abdullah, A., & Alzain, M. A. (2021). A load balancing algorithm for the data centres to optimize cloud computing applications. IEEE Access, 9, 41731-41744.
42. Shahapure, N. H., Rekha, P. M., & Poornima, N. (2021). Threshold Compare and Load Balancing Algorithm to for Resource Optimization in a Green Cloud. Revista Geintec-Gestao Inovacao E Tecnologias, 11(4), 4465-4481.
43. Thakur, A., & Goraya, M. S. (2017). A taxonomic survey on load balancing in cloud. Journal of Network and Computer Applications, 98, 43-57.
44. Ullah, A., Nawi, N. M., Uddin, J., Baseer, S., & Rashed, A. H. (2019). Artificial bee colony algorithm used for load balancing in cloud computing. IAES International Journal of Artificial Intelligence, 8(2), 156.
45. Volkova, V. N., Chemenkaya, L. V., Desyatirikova, E. N., Hajali, M., Khodar, A., & Osama, A. (2018, January). Load balancing in cloud computing. In 2018 IEEE conference of russian young researchers in electrical and electronic engineering (EIConRus) (pp. 387-390). IEEE.
46. Wen, W. T., Wang, C. D., Wu, D. S., & Xie, Y. Y. (2015, August). An ACO-based scheduling strategy on load balancing in cloud computing environment. In 2015 Ninth international conference on frontier of computer science and technology (pp. 364-369). IEEE.