applied sciences

MDPI

Article

Performance Evaluation of Load-Balancing Algorithms with Different Service Broker Policies for Cloud Computing

Muhammad Asim Shahid [1], Muhammad Mansoor Alam [1,2,3,4] and Mazliham Mohd Su'ud [4,*]

1 Malaysian Institute of Information Technology, Universiti Kuala Lumpur, Kuala Lumpur 50250, Malaysia
2 Faculty of Computing, Riphah International University, Sector I-14, Hajj Complex, Islamabad 46000, Pakistan
3 School of Computer Science, University of Technology Sydney, 15 Broadway, Ultimo, NSW 2007, Australia
4 Persiaran Multimedia, Multimedia University, Cyberjaya 63100, Malaysia
* Correspondence: mazliham@mmu.edu.my

Abstract: Cloud computing has seen a major boom during the past few years. Many people have switched to cloud computing because traditional systems require complex resource distribution and cloud solutions are less expensive. Load balancing (LB) is one of the essential challenges in cloud computing used to balance the workload of cloud services. This research paper presents a performance evaluation of the existing load-balancing algorithms which are particle swarm optimization (PSO), round robin (RR), equally spread current execution (ESCE), and throttled load balancing. This study offers a detailed performance evaluation of various load-balancing algorithms by employing a cloud analyst platform. Efficiency concerning various service broker policy configurations for load-balancing algorithms' virtual machine load balance was also calculated using metrics such as optimized response time (ORT), data center processing time (DCPT), virtual machine costs, data transfer costs, and total cost for different workloads and user bases. Many of the past papers that were mentioned in the literature worked on round robin and equally spread current execution, and throttled load-balancing algorithms were based on efficiency and response time in virtual machines without recognizing the relation between the task and the virtual machines, and the practical significance of the application. A comparison of specific load-balancing algorithms has been investigated. Different service broker policy (SBP) tests have been conducted to illustrate the load-balancing algorithm capabilities.

Keywords: cloud computing; load-balancing algorithms; parameters; service broker policies; virtual machines

check for updates

Citation: Shahid, M.A.; Alam, M.M.; Su'ud, M.M. Performance Evaluation of Load-Balancing Algorithms with Different Service Broker Policies for Cloud Computing. *Appl. Sci.* **2023**, *13*, 1586. https://doi.org/10.3390/app13031586

Academic Editors: Mingyang Guo, Jiang Long, Qingdong Li and Zun Liu

Received: 29 November 2022
Revised: 5 January 2023
Accepted: 11 January 2023
Published: 26 January 2023

1. Introduction

Because of the exponential growth of cloud computing over the last 10 years, multiple corporations and business sectors have transitioned to the cloud to ensure transparency, scalability, and accessibility [1]. Most people are switching to the cloud due to cost savings and complex resource distribution. In a cloud environment, the corresponding infrastructure is managed and handled by cloud providers [2]. However, several challenges in cloud computing require resolution before the true implementation of the cloud. Among them, resource management or load balancing in cloud computing has become a big challenge. Load balancing is a technique for spreading the burden on several machine assets via network connections to achieve optimum resource efficiency and minimal data analysis time, and to prevent overloading. Load balancing implies that at any moment all processors in the system, as well as in the network, will perform almost the same amount of work [3]. Different load-balancing algorithms have been discussed in the literature.

This study provides a performance evaluation of various load-balancing algorithms in a cloud computing environment. For this purpose, a cloud analyst has been used for the implementation. There are several analysis simulators available that can be used for testing environments and these tools support testing the output of scaling apps on the

Internet [4]. This paper has added load-balancing algorithms into cloud analyst simulators and performed the analysis with different service broker policies in the cloud environment using CloudAnalyst. Load-balancing algorithms can be implemented in various sectors such as artificial neural networks, function optimization, and fuzzy control of the system. Additionally, they can be used to schedule a straightforward workflow in cloud-based virtual machines [5].

Organization. The rest of the paper is organized as follows. In the next section, we put our work into context concerning related work, and Section 3 is based on research methodology. In Section 4, we present our experimental results, and Section 5 is based on the discussion. We conclude in Section 6.

2. Related Work

There has been a lot of research conducted on cloud computing, cloud analysts, virtual machine load balancers with different service broker policies, and so on. Table 1 is a summary of the literature on LB algorithms in cloud environments.

Table 1. A summary of the literature on load-balancing algorithms in cloud computing environments.

Ref.	Author Name	Year	Pros and Cons	
[1]	Ahmed I. El Karadawy et al.	2020	Provided the nearest data center strategy with optimal average response time (milliseconds).	Does not provide a virtualized cloud environment with data center execution time.
[2]	Sunny Nandwani et al.	2015	Presented the results of ORT, data center processing time, and cost.	It does not show response time by region of LB policies.
[3]	S. Suguna and R. Barani	2015	Every algorithm is analyzed and its scheduling parameters such as average response time, service time for the data center, and overall cost of the various data centers are identified.	Lack of heterogeneous environments due to handling big data and need to improve overall response time with reduced cost.
[4]	Simar Preet Singh et al.	2016	• Displaying the performance of the various LB algorithms. • As compared to RR and ESCE LB techniques, throttled's reaction time is good for six data centers and six user bases.	• All of these algorithms have the same cost. • The various PSO strategies are not being used.
[5]	Divyani et al.	2018	• RR has the maximum response time. • The greatest result against all parameters.	• Throttled has a substantially lower response time. • RR does not produce an extra effective outcome.
[6]	Ritesh Patel and Sandip Patel	2018	The number of virtual machine relocations is managed as a result of considering the dynamic host-to-virtual-machine proportion.	• Failure to modify spending consequences. • Even so, by taking into account another parameter, the rate of free resources, the outcomes can be improved.
[7]	Ahmed M. Manasrah et al.	2017	• Reduced the processing and response time of client needs in an appropriate cost range. • A good collection of data centers.	• The data center does not take into account the work size. • The throttled strategy is not being used.

Table 1. *Cont.*

Ref.	Author Name	Year	Pros and Cons	
[8]	Meeta Singh et al.	2018	• The RR process time is optimum. • Improves the performance of the available resources and evenly distributes the load across the servers.	The processing time of throttled data is very short.
[9]	Sohaib and Abdul Razzaque	2017	RR is optimal with the average response time.	• Lack of difference between the costs. • Throttled and ESCE algorithms are not providing the best average response time as compared to RR.
[10]	Imtiyaz Ahmad et al.	2017	The throttled algorithm is organized in nature.	• Lack of maximum time. • There is a lack of knowledge about virtual machines and data center overcrowding.
[11]	Amrita Jyoti et al.	2020	• More metrics are considered, and the evaluations are taken into account. • Capable of measuring current LB strategies.	• Lack of virtual machine migration. • Lack of service broker policies. • Lack of server.
[12]	Hetal and Ritesh	2015	• Included a cloud infrastructure that is geographically dispersed. • This recommends a new framework for service brokers that will increase either costs or efficiency.	• CloudSim is narrowed to virtual machine management. • Virtual machine management is not easy to modify.
[13]	Zakaria Benlalia et al.	2019	• A novel service broker algorithm for cost and response time optimization has been developed. • The suggested policy selects the best data center based on the efficiency/cost ratio. If the data center with the lowest ratio is available, it is selected; otherwise, the closest data center is selected.	• In CloudAnalyst, there is no comparison with other policies. • The proposed algorithm has not been integrated into the Cloud Analyst simulator.
[14]	M. Radi	2015	The average overall response time presented a huge improvement with current policies.	Different configurations and virtual machine load balancers are not providing the proposed policy.
[15]	Shivam Chugh et al.	2015	Out of the three service broker policies, the parameters measured, i.e., response time and DCPT are the lowest of the closest data center policy.	Every cloud engineer faces a challenge when it comes to building a network because of response time and DCPT.
[16]	Reza Mesbahi et al.	2016	• The threshold algorithm performs faster. • The threshold has provided a virtual machine list.	• RR fails to deliver an additional good output. • ESCE also fails to deliver additional better outcomes.

Table 1. *Cont.*

Ref.	Author Name	Year	Pros and Cons	
[17]	Rajeshwari Nema et al.	2016	• RR requires little time to execute SBP. • The active monitoring LB strategy is dynamic. • In terms of efficiency and response time, the throttled algorithm is ideal.	• RR assigns the request directly (such as whether it is over-loaded or not) without verifying the existing server capability. • AMLB will always consider the least-loaded virtual machine for assigning new incoming requests, but will not verify whether or not it has been used recently (so some virtual machines are over-used and some are still suitable). • In the throttled algorithm, each time the data center requests a load balancer for a virtual machine it assigns the index table.
[18]	Ojasvee and R K Banyal	2016	• Reduces the number of free host processors. • Evaluates the highest number of processors and allocates new entrant procedure to the host.	There are no free hosts for the next data center.
[19]	Sandip Patel et al.	2015	• The RR LB scheme provided requests from the data center in a centralized way. • When compared to the previous two policies, the throttled algorithm performs the best in terms of performance and response time. Its behavior is likewise dynamic.	• Does not include the preceding allotment of a virtual machine to an application. • RR assigns requests without first assessing the server's capacity (such as whether it is overloaded or not). • AMLB will always choose the least-loaded virtual machine to assign new incoming requests to, regardless of whether it has been used before (so some virtual machines are over-utilized and some are still ideal).
[20]	Shodhganga	2015	• RR is easy to deploy. • Throttled is easy to deploy. • ESCE is easy to deploy.	• Organized and static, so RR is not appropriate for cloud environments. • Throttled waiting time is usually huge. • ESCE is centralized.
[21]	Shalini and Uma Kumari	2016	• Fulfill the user's requirements. • Enhance resource use. • Improves system performance.	• Lack of elasticity. • Lack of assigned resources.
[22]	Komalpreet and Rohit	2018	• ESCE provides better safety. • ESCE is capable of tackling failures.	• ESCE is centralized. • ESCE also fails to deliver additional better outcomes.
[23]	Patel and Chirag	2015	• Achieved a better level of fault tolerance. • Achieved greater scalability. • Efficient use of resources.	• The length of process execution is not mentioned. • Does not take into account the present situation.

Table 1. *Cont.*

Ref.	Author Name	Year	Pros and Cons	
[24]	Ankit and Mala Kalra	2016	Based on their priority, status, and memory use, the suggested algorithm distributes the load across the virtual machines in an appropriate manner.	The proposed approach ignores the virtual machine's energy awareness and reliability.
[25]	Reena and Bhawna	2015	• Effectively, the response time has enhanced. • A load of service requests could be distributed efficiently through virtual machines.	The suggested algorithm's fundamental flaw is that it checks the availability of all virtual machines every time it assigns a new load. As a result, request allocation takes longer, resulting in a longer response time.
[26]	Mazen Farid et al.	2020	• The scheduling system lowers the makespan and expense of operation. • Scheduling also increases the availability of resources and device scalability for cloud providers.	• Lack of task scheduling in the heterogeneous cloud environment. • Lack of understanding of the various reliability rates.
[27]	Anurag and Rajneesh	2016	The proposed strategy suggested is ideally suited to cloud surroundings.	Lack of information concerning the overloading of virtual machines and data centers in advance.
[28]	Kalpana and Y Rama Devi	2015	• Analysis of various simulators used for cloud computing. • The emulator has features that distinguish it from others.	CDOSim, TechCloud, DCSim, and GroudSim are not providing the platform, networking, simulation type, or open-source availability.
[29]	Utkal and Mayank	2015	• GUI Assistance, underneath the programming language. • CloudSim, SPECI, and GreenCloud are platform-supported simulators.	• Lack of choosing a suitable simulator according to user needs. • GroudSim is not providing the platform, networking, simulation type, or open-source availability.
[30]	Amrita and Manish	2020	Multi-agent deep reinforcement learning to dynamic resource allocation is used to decrease the issue of cloud service.	The proposed solution does not include a dynamic approach based on the virtual machine's cost analysis.
[31]	Kalka Dubey et al.	2020	The proposed policy increased the overall execution time and speed of resource usage.	The proposed policy has the issue of increased LB issues related to power consumption.
[32]	Archana and Rakesh	2020	• Give optimal response time and DCPT. • When compared to the existing RR and evenly distributed execution technique, throttled takes less time.	• Failure to validate which algorithm is ideal. • The response time provided by these algorithms is not optimal.
[33]	Ashmeet and Meenu Dave	2019	• RR has the maximum response time. • The throttled algorithm is organized in nature. • ESCE is easy to deploy.	• Throttled is substantially lower. • RR does not produce an extra effective outcome.
[34]	Al-Tarawneh and Amjed	2019	• The proposed algorithm can accomplish better changes in performance. • It can also work in a user-oriented approach.	• Lack of security ability and security criteria for using apps. • The proposed algorithm lacks consumer expenses and contributor income.

Table 1. *Cont.*

Ref.	Author Name	Year	Pros and Cons	
[35]	Sweekriti and Sudheer	2019	• Well-organized resource usage. • Better organized than RR, throttled, and ESCE LB algorithms.	• The proposed algorithm lacks live migration of virtual machines. • The proposed algorithm lacks auto-scaling strategies.
[35]	R. Valarmathi and T. Sheela	2019	• Various routing schemes are in sequence to decrease the processing time. • The data center is used to execute the task based on the limited distance of the track.	The proposed algorithm lacks improvement in minimal and maximal processing time.
[36]	Elena and Almothana	2019	This research paper tests the efficiency of the three current service broker policies with a strategy of sustained load leveling.	The research paper lacks analysis of the closest data center (CDC) and reconfigures dynamically with load policy.
[37]	Pawan and Rakesh	2018	The proposed algorithm distributes tasks effectively among virtual machines based on existing load, priority, memory use, and state of the virtual machines.	The proposed algorithm is not providing user requirements among the virtual machines based on the reliability and use of the processor.
[38]	Swati and Saurabh	2018	• Job scheduling is about reducing the makespan. • The effectiveness of the proposed algorithm is significantly expanded.	The proposed algorithm has a lack of demand for increasingly more jobs.
[39]	Rajeshwari and Sahana	2016	Enhanced RR is proposed, and findings demonstrate that it outperforms other existing algorithms in terms of average request service time.	Lack of improvement in active monitoring and throttled LB algorithm.
[40]	Slesha and Pragnesh Patel	2015	The throttle algorithm decreases response time, data center request servicing time, and cost.	The throttled algorithm lacks virtual machines that possess various hardware configurations.
[41]	Hemant S. Mahalle et al.	2015	The proposed algorithm is providing the closest data center and maximize response time functions with low processing times effectively.	The proposed algorithm is not providing extra giant outputs for a cloud computing domain.
[42]	Anant Gaur and Kush Garg	2015	The proposed algorithm has service broker approaches that decrease response time and overall data center costs.	The proposed algorithm is not providing data center configurations using the different broker policies.
[43]	Almothana Khodar et al.	2020	The cloud computing key benefit is that consumers are liberated from fears about learning simple instrumentality that is sensitive to inquiries.	The allocation of resources, the availability of resources that suit requirements, and a lot of cloud-faced box management problems.
[44]	Soumya et al.	2020	The researcher provides the communication model with the CloudAnalyst platform.	The communication model for CloudSim is restricted.
[45]	Sasmita and Bibud-hendu	2020	The novel broker policy reduces overall costs, response time, and processing.	The novel broker policy is not using optimized methods for cost and response time.
[46]	V. Arulkumar and N. Bhalaji	2020	The water wave algorithm performs better concerning total response time.	The water wave algorithm has delayed various routing policies.
[47]	Anand Nayyar	2016	The cloud analyst is a convenient-to-use GUI for establishing and viewing the results of cloud computing experimentation of all varieties.	The cloud analyst has security challenges.

Table 1. *Cont.*

Ref.	Author Name	Year	Pros and Cons	
[48]	Khaled M. Khalil et al.	2017	All simulators are provided as a service to end-users over a network.	The research paper said that cloud computing is very costly and difficult at the actual Internet site.
[49]	Khadijah Bahwaireth et al.	2016	The research paper discussed CloudSim, CloudAnalyst, and CloudReport, as well as CloudExp, iCanCloud, and GreenCloud, which are among the tools highlighted.	The research paper lacks massive cost measurement for configuration and execution.
[50]	Pericherla	2016	• The proposed algorithm provides minimum cost. • The proposed algorithm has quality that is repeatable and checkable. • The proposed algorithm has environmental consistency.	Due to the high amount involved in establishing a cloud, conducting research on live cloud environments for individuals or small organizations is extremely difficult.
[51]	Fairouz Fakhfakh et al.	2017	• The research paper said cloud computing provides on-demand computing tools and services. • CloudSim, NetworkCloudSim, FederatedCloudSim, and DynamicCloudSim are open source.	• Constraints on QoS. • Experimentation is a difficult issue in a real environment. • CloudSim, NetworkCloudSim, FederatedCloudSim, and DynamicCloudSim are not providing graphical support or SLA support.
[52]	Muhammad Asim Shahid et al.	2020	• The research paper meets the deadline for the makespan. • The study report also looks at the issues with LB in the cloud computing context and the necessity for a new LB algorithm that uses fault tolerance measures.	The research paper policy is unable to predict when the burst will occur.
[53]	Muhammad Asim Shahid et al.	2021	Fault tolerance is a feature of LB algorithms, which means the author can provide standardized LB despite arbitrary node or connection errors.	In this research paper, the lack of the availability of key resources, as well as the installation of applications, are a concern.

Ahmed I. El Karadawy et al. [1] discussed a comprehensive analysis of the cloud analyst simulator on various algorithms of LB with different service broker policies. Specifically, they analyzed three distinct LB algorithms: RR, throttled, and ESCE. Sunny Nandwani et al. [2] discussed the numerous and current service broker policies and LB algorithms. These LB algorithms were also compared with various service broker policies and simulations on cloud analysts to check the execution of existing algorithms; using these can compare the performance based on different metrics. S. Suguna and R. Barani [3] suggested that two LB algorithms, the ESCE algorithm and throttled algorithms, are used by cloud analysts to evaluate the accuracy of the algorithms. Simar Preet Singh et al. [4] suggested the cloud analyst simulator and the importance of cloud architecture infrastructure services. Divyani et al. [5] carried out simulation studies for the performance of different variations of LB and service broker policies, and also developed a new dynamic LB algorithm to integrate the key qualities of other policies.

Ritesh Patel and Sandip Patel [6] this research paper proposed the problem of data centers and services that could meet demand and decided on a few issues, such as higher prices and under-use of assets. Furthermore, they proposed an effective performance data center algorithm that determines the data center and resource to host ratios and has unlimited available resources, which is dependent on virtual machines in a similar direction. Ahmed M. Manasrah et al. [7] proposed a variable service broker routing policy, which is a heuristic-based strategy that aims to accomplish minimal response time by recognizing the throughput, bandwidth, and task scale of the communications platform. The proposed service broker strategy would also decrease data center congestion by diverting client requests to the next data center which would provide a good response and process time. Meeta Singh et al. [8] concentrated on LB to boost the effectiveness of the existing resources and allocate the load uniformly around the servers. Sohaib and Abdul Razzaque [9] suggested a wide range of distributed environment LB issues which are evolving and attempting to achieve focus. Additionally, they conducted a comprehensive analysis of the RR virtual machine load balancer based on efficiency comparisons. Imtiyaz Ahmad et al. [10] proposed a simple throttled mapping approach between tasks and resources for the user to enhance the performance of the current cloud system.

Amrita Jyoti et al. [11] proposed the service broker policies intending to achieve reliability and scalability, reduce response time, and optimize cloud system efficiency and costs. They also discussed the LB algorithms and the analysis of service broker policies. Hetal and Ritesh [12] discussed the cloud simulators and service broker policies and explained the problems of service broker policies to correct cloud analysts that focused on data centers. Zakaria Benlalia et al. [13] proposed the software and hardware infrastructure, data center selection, and service broker policies and proposed a new SBP by combining the effectiveness of performance to maximize cost and response time. M. Radi [14] suggested that LB algorithms achieve high customer satisfaction and strategic use of cloud computing and data center selection. Additionally, the proposed service broker policies were compared with three existing service broker policies regarding the overall average response time utilizing various LB algorithms for the virtual machine. Shivam Chugh et al. [15] indicated a complete comparison with the data center process time in a single data center for various service broker policies. Simulation results were also given based on the RR scheduling algorithm applied to various service broker policies to measure response time and processing time for data centers.

Reza Mesbahi et al. [16] suggested the distributed system burden and managed all incoming requests in cloud computing environments across all processing nodes. Additionally, they addressed service broker policies in cloud computing and empirical analysis was presented for the configurations of virtual machine LB algorithms. Rajeshwari Nema et al. [17] discussed the cloud computing and LB problems and proposed an algorithm to improve the overall efficiency and supply the demanding customers with greater satisfaction. Additionally, they applied a new approach to current LB policies. Ojasvee and R K Banyal [18] proposed the efficient use of cloud resources and increased accessibility by changing the basic LB algorithms and allocating virtual machines to user bases using the cloud bus and CloudSim simulator. Sandip Patel et al. [19] indicated that the LB problems and current algorithms should aim to provide optimized frameworks and delegate the customers' demands to cloud nodes that are active, as well as seek to improve the cloud's overall efficiency and provide more customer interaction and good service. Additionally, they conducted an investigation of different LB policies in cloud analysts with pros and cons. Shodhganga [20] proposed that LB algorithms at multiple rates would lead to better outcomes than those produced by combining the best characteristics of various LB strategies.

Shalini and Uma Kumari [21] suggested that cloud computing architecture, virtualization, and LB issues with different LB algorithms are currently available in cloud computing. Komalpreet and Rohit [22] suggested that an ESCE algorithm has been contrasted with the RR, and that the proposed data center efficiency in terms of cloud settings can be achieved successfully. Patel and Chirag [23] suggested the different LB algorithms and indicated how these algorithms help to solve the problems of load transfer between various virtual machine resources. Ankit and Mala Kalra [24] suggested that the LB algorithm efficiently allocates the incoming jobs in the cloud data center between virtual machines and incorporated using the cloud analyst simulation tool in this research paper, and the performance of the proposed technique was evaluated with the three current existing algorithms based on response time. Reena and Bhawna [25] indicated using the dynamic load management algorithm for effectively distributing the current incoming request among the virtual machines. Additionally, the results were implemented using a cloud analyst simulator based on different metrics, such as data processing time and response time, etc., and they compared the results with the previously developed virtual machine algorithm.

Mazen Farid et al. [26] proposed that the study of scheduling algorithms be focused on the enhancement of PSO. This was intended to help people decide on the most appropriate quality of service considering massive processes in infrastructure of the service, cloud apps, and resource mapping tasks. Anurag and Rajneesh [27] indicated the two-level load balancer framework and showed a two-level load balancer model by combining the idle queue and shortest queue approaches. A cloud analyst simulator was used to evaluate the proposed framework. Kalpana and Y Rama Devi [28] suggested the study of analysis with a comparison of different current cloud simulator software. Utkal and Mayank [29] suggested the characteristics of the current cloud computing modeling. Additionally discussed was the cloud computing simulation tool. Amrita and Manish [30] referred to the vast number of service demands from users and suggested increasing the difficulty of the network infrastructure. Additionally, they implemented a novel method for producing dynamic LB and service brokering-based resource provisioning.

Kalka Dubey et al. [31] suggested a novel virtual machine allocation strategy called the efficiency of service best fit decreasing. As a result of simulation tests, the proposed policy results were much better than first-fit and best-fit policy results. Archana and Rakesh [32] proposed studying the three cloud LB algorithms, namely RR, throttled, and ESCE, and the output in terms of average response time, hourly data center response time, and virtual machine value, etc., using the cloud analyst simulator. Ashmeet and Meenu Dave [33] proposed the LB algorithm to increase app performance, as well as explained a comprehensive evaluation of LB algorithms using the cloud analyst simulation platform. Al-Tarawneh and Amjed [34] indicated an adaptive cloud SBP based on fuzzy cloud services. Additionally, the proposed adaptive fuzzy-based framework to find the most suitable data center for consumer cloud service demands took into account the quality and cost expectations of users. Sweekriti and Sudheer [4] suggested that the modified central load balancer algorithm keeps away from the burden and loading of virtual machines. Using the CloudSim simulator, the MCLB algorithm compares it with the current RR, throttled, and ESCE algorithms.

R. Valarmathi and T. Sheela [35] proposed the specific service broker approach for planning to rely on positioning the tasks delivered in the data center with minimal route duration and minimal equipment. The key goal was to achieve a limited response time depending on the means of transmission, bandwidth, latencies, and task length. Elena and Almothana [36] conducting a survey and evaluating various current service broker policies and load-equalizing algorithms in cloud computing and various simulation applications, and testing the fulfillment of existing algorithms, such as the Cloud Analyst. Pawan and Rakesh [37] proposed that the LB algorithms assign the jobs of virtual machines to be based on the virtual machine's load and utilization. Additionally, they applied and analyzed the proposed algorithm to a cloud analyst simulator. Swati and Saurabh [38] suggested the modification of this strategy by implementing a novel scheduling algorithm that could enhance the response time and handle the stability of the load. The assessment was based on the cloud analyst simulator, and findings confirmed the proposal's efficiency which could minimize response time and average processing time. Rajeshwari and Sahana [39] proposed an algorithm for the enhancement and optimization of cloud improvement. The assessment was based on the cloud analyst simulator.

Slesha and Pragnesh Patel [40] suggested that a comparative analysis should be conducted for the current throttle algorithm and proposed the improved throttled algorithm for LB in cloud computing. The proposed throttled algorithm was put into practice and tested. These two algorithms were differentiated using the cloud analyst simulator in terms of response time, data center request time, and cost. Hemant S. Mahalle et al. [41] the research paper explains the cloud and proposes that assets in cloud computing with application service broker policies would have huge results. Anant Gaur and Kush Garg [42] indicated the algorithm and some novel methods which could be used to achieve significant enhancement in the response time of the method in contrast to conventional methods. The assessment was based on the cloud analyst simulator. Almothana Khodar et al. [43] suggested the analysis and evaluation of current varied service broker policies and load equalization algorithms in cloud computing. Soumya et al. [44] suggested better use of some of the essential open-source cloud toolkits for comparative analysis. They made subtle changes to the code that will be very useful for creating new calculations for all of the users involved in various processes and designs.

Sasmita and Bibudhendu [45] proposed a new strategy for service brokers to decrease the overall cost. The overall cost reflects the cost of the virtual machine and the cost of data transmission. The method proposed also reduces the data center response time and data center processing time. V. Arulkumar and N. Bhalaji [46] suggested that LB algorithms are inspired by the efficient measurement of response time and DCPT in a cloud environment. The simulation was conducted using a cloud analyst simulator. Anand Nayyar [47] indicated the best open-source cloud computing simulators and described the benefits of cloud simulators as well. Khaled M. Khalil et al. [48] studied the 33 cloud simulators with cloud architecture and conducted a test of these simulators on various parameters. Khadijah Bahwaireth et al. [49] suggested the most effective simulation tools in this study. This included CloudSim, CloudAnalyst, CloudReports, GreenCloud, CloudExp, and iCanCloud, and they conducted experiments to display the functionalities.

Pericherla [50] indicated that several cloud simulators that have been developed over the years are a cost-effective way to perform cloud research work. This work also compares nearly 17 cloud simulators based on a variety of factors and presents the results and justifications allowing new researchers to choose an appropriate cloud simulator. Fairouz Fakhfakh et al. [51] suggested the available tools in cloud computing for simulation. This work also offers a basic and comparative overview of the methods under review. Finally, a key obstacle to tackle in further studies stands out. Muhammad Asim Shahid et al. [52] proposed that existing LB techniques also explore the challenges of LB in the cloud computing environment and identified the necessity for a novel LB algorithm that uses fault tolerance metrics. Muhammad Asim Shahid et al. [53] proposed that the major goal of LB is to efficiently control the load across numerous cloud nodes such that the node is under- or overloaded. LB can help you save time and cost, and improve the performance of the system. LB entails optimizing the entire system's performance.

3. Research Methodology

In this section, the paper's contribution is the research methodology. After discussing the research methodology, the next section discusses the results and findings and this paper's contribution will be compared with the state-of-the-art techniques currently used by cloud analysts. There are clear pros and cons of these LB algorithms that are shown in Table 2.

3.1. Particle Swarm Optimization

PSO follows the normal action of flocking birds and was created by Kennedy and Ederhart in 1995. Typically, birds swarm when they see a spot to feed. The birds' natural behavior is used in the cloud to find the optimum virtual machine for handling the burden. The bird is thought to fly in the cloud space and show natural flocking behavior, while obtaining food selects the best remedy to make the virtual machine manage the situation. Each task discovers its best local virtual machine and also keeps the information about the best global virtual machine and the lowest appropriate route recognized by the task at the moment. The task changes its velocity in each step depending on the optimum location of the best task in the incoming overall task. The optimal workplace (task allocation to the virtual machine) is calculated from both local best and global best. The variations proceed until it finds a global solution [46].

Table 2. Pros and cons of the load-balancing algorithm.

Strategy	Pros and Cons	
Round Robin	• RR does not demand contact between processes [10]. • The efficiency of the algorithm is assessed using the cloud analyst simulator, and the results are compared to those of the RR and throttled algorithms. As a result, the response time improves in comparison to the previous one [25]. Broadly adopted and effective in deployment [26]. • It does operate circularly [39].	• Between virtual machine mapping tasks, it does not take into account the size of the task nor does it recognize the virtual machine's processing ability. It does not find the existing load on the virtual machine either [25]. • On the condition that the server has batch processing capabilities, overloading and failure can occur [26]. • If the virtual machine has been delegated to an application for a user program, its status will not be retained [41].
Equally Spread Current Execution	• ESCE algorithm can return the ID to the data center controller, and the data center controller can then send a request to the virtual machine indicated by that ID. The data center then notifies the ESCE of the new request allocation [25]. • Effective for both tiny and static contexts [25]. • Improves load times and response times at the data center [28].	• Required to increase the time taken to process the data [28]. • It would not be fault-tolerant, and it also has a single point of malfunction issue [29].
Throttled	• The throttled algorithm is an LB approach that evenly distributes incoming job requests among servers or virtual machines [25]. • No lines are left at the stage of the virtual machine. Just one queue is held on the scheduler stage [25]. • The throttle LB algorithm has the perfect LB solution [47].	• Naturally centralized [25]. • The waiting period is normally substantial [25]. • Will not take into account the size of the job and server power through the project map file [25].

The LB algorithms add to a cloud analyst simulator. The PSO is provided that solves the problem of loading balance of virtual machines to effectively create corresponding relationships between tasks and virtual machines. Aiming at finding an almost optimized scheduling solution not only makes the task execution time the shortest but can make the resource utilization of virtual machines the maximum possible. The PSO algorithm involves numerous metrics such as simulation duration, number of regions user bases, data center, number of virtual machines, user grouping factor in user bases, data center request grouping factor, executable instruction length per request, etc. [46].

The processing time varies according to the performance of the data center such as RAM, CPU, and arrangement of the virtual machine. The timeframe required to process the job will rely on the computing task to be performed. For example, a simpler job needs limited processing time if no input–output process has been involved. As a result, the PSO provides the ORT with a slow response, but the CDC reconfigures dynamically with a load having the maximum response time. The results of the simulation show that the algorithm has a fast convergence speed, high efficiency, and practical significance for the application [46].

3.2. Round Robin

This is one of the simpler schedule techniques which uses the time-slicing concept. Here, the duration is split into several segments, and each node is assigned a different time-slicing or time range, i.e., it uses the timing schedule rule. Every node is assigned an action and a quantum [18]. Time quantum is the main consideration for LB algorithms [33]. The RR algorithm is the same as the first-come-first-served algorithm. Furthermore, when the time quantum is insufficient, the RR algorithm is referred to as a processor-sharing algorithm [17]. The service providers provide services to the requested customer on a time-slicing plan [18]. The procedure of virtual machines running in an RR model is depicted in Figure 1 [1].

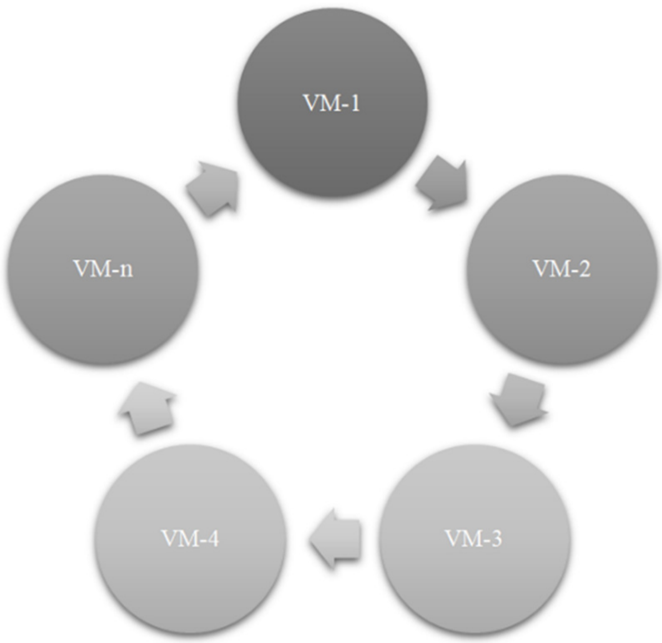


Figure 1. The relationship between virtual machines in the RR algorithm [1].

3.3. Equally Spread Current Execution

This algorithm uses the concept of a distributed spectrum and operates in such a form that the number of available activities in each virtual machine is equal at any moment. The scheduler retains a virtual machine allocated table that holds the virtual machine's IDs and counts of the virtual machine for current tasks. Active task counts belonging to that virtual machine in the allocations tables will be changed with the assignment of new tasks or on fulfillment of tasks. An active task count for each virtual machine is 0 at the start. Upon entry of the mission, the ESCE scheduler notices a minimum virtual machine that is engaged in mission numbers. If more than one virtual machine has the lower active counting then the first listed virtual machine for job assignment is chosen. Task queues shall be controlled according to each virtual machine [19]. Figure 2 depicts the process of assigning a new virtual machine from the data center controller; the strategy selects a virtual machine from the index with the lowest load value (if the strategy detects two or more virtual machines with the same value, it selects the first detected virtual machine), and then the strategy sends the selected virtual machine ID to the data center controller [1].

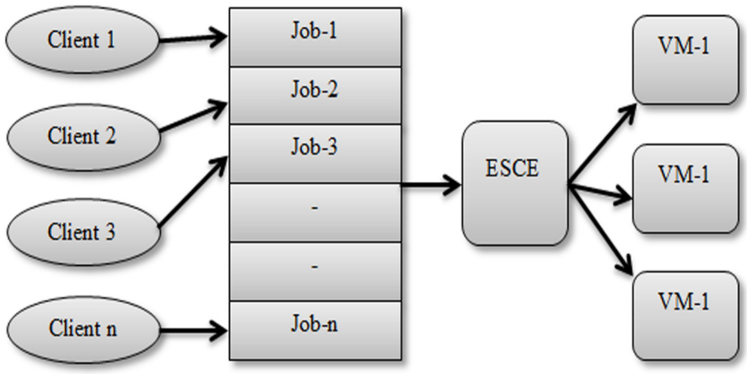


Figure 2. The process of the ESCE algorithm [1].

3.4. Throttled

The throttled algorithm is better measured in terms of efficiency and response time to current dual strategies [20] in heterogeneous cloud computing [38]. It is also behaviorally complex. Efficiently, it delegates all incoming work to a virtual machine. This explores the correct virtual machine for allocating a particular task to. With this indexed collection, the work administrator has a number of all virtual machines which allocate the desired work to the correct machine which can easily identify the load and fulfill the activities. If the work is very well suited to a specific virtual machine concerning the size and availability of the virtual machine and that work is allocated to the correct machine, then the task administrator waits for the request of the customer and places the job in the list for quick execution. This algorithm works well in contrast with the RR algorithm [20]. Figure 3 depicts the procedure used in the throttled method.

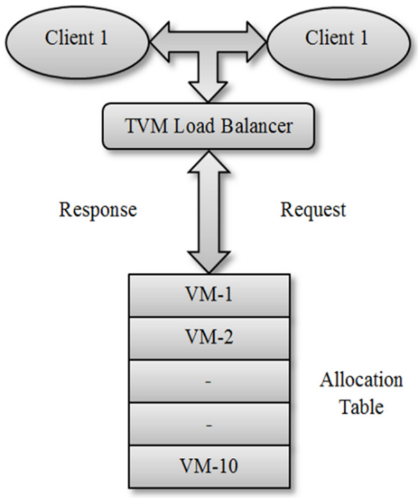


Figure 3. The process of the throttled algorithm [1].

3.5. A Comparative Study of Service Broker Policies

A service broker determines that the data center will supply the support for customers' demands. Additionally, the service broker manages the flow of data through consumer and data centers [12]. The service broker acts above this stage as a moderator between the customers and cloud service providers [42]. SBP distributes the services dynamically between the cloud infrastructure and cloud service providers [36] or, in plain terms, it is the choice strategy for the data center [12]. The layout of the virtual machine to the physical machine in the data center is termed virtual machine deployment and is a very critical part of the data center broker [37]. One could consider what kind of situation the SBP operates in. The figure shows certain data centers and users. After demand from the customer, the SBP helps to determine which data center will be offering the service for potential demand [12]. Figure 4 displays the data centers which are spread geographically [13].

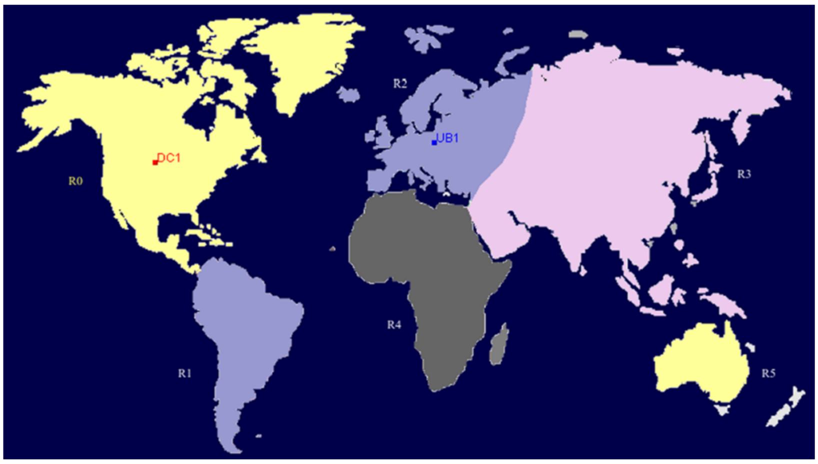


Figure 4. The geographically distributed data centers [12].

3.6. Categorization of Service Broker Policies (SBPs)

The service broker policies are used to determine which data center serves the requests from each user to control the routing bottleneck problem between the data center and user bases. The service broker policies consist of some broker policies which help to provide a data center for an upcoming request. They also offer a special, standard GUI from which consumers can distribute and control their operations throughout several clouds, since this feature is also defined as the selection policy for the data center. The service-level agreement (SLA) must be defined and the customer must be provided with a standard GUI to control and track the managed services. The broker process can be interconnected to cloud computing systems and can involve more than one acceptable physical cloud computing tool designed to perform each and more broker operations [11].

The newest version of cloud analyst incorporates three various kinds of service brokers, each with its unique routing strategy [14]. Figure 5 indicates the policies for the service brokers [24]. There are clear pros and cons of these service broker policies that are shown in Table 3.

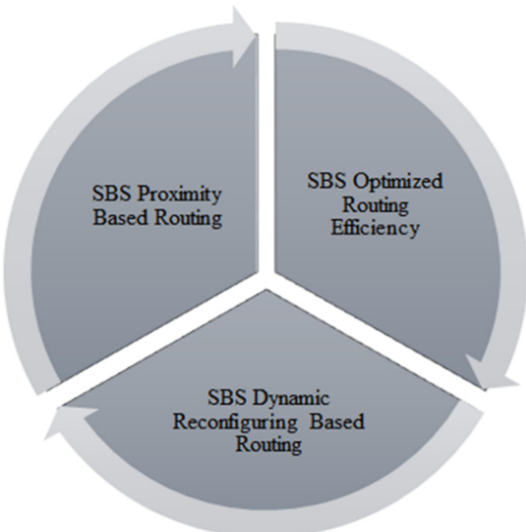


Figure 5. The categorization of service broker policies [14].

Table 3. Pros and cons of service broker policies in cloud computing environments.

Policy	Pros and Cons	
Proximity-Based Routing	• It is one of the most straightforward methods for simulating a cloud environment [1]. • This policy is known as the data center with the shortest path [14].	• The difficulty with this method is that some virtualized data centers become overwhelmed while others remain underutilized [1]. • It redirects the demands to other service brokers if there are no matching results between the user's demands [11].
Optimized Routing Efficiency	• In this method, the service broker observes all of the required attributes in the data centers to assess their performance [1]. • This policy gives the end-user the fastest response time possible at the time of query [14].	• Routing based on proximity to services selects the quickest path from the data center to user requests, which may cause the nearby data center to become overburdened because the channel bandwidth is really limited [11]. • The optimum path is chosen by performance-optimized routing, which is dependent on the latency network and data center workload. As a result, time is wasted [11].
Dynamic Reconfiguring-Based Routing	• The service broker is in charge of delivering scalability to the cloud system's applications [1]. • The service broker adjusts the number of virtual machines in the data centers dynamically [14].	• There is no support for diverse cloud service models, and there is no desirable support for QoS features with these service broker techniques [11]. • Achieves a time of execution of 0 s [11].

3.6.1. SBP Proximity-Based Routing [14]

This policy is called the data center with the shortest route. In light of network bandwidth, the operation brokers transfer the query to the nearest data center [15]. This policy provides a network, latency-based, dynamically ordered list of data centers [40]. If more than one data center exists inside the same region, it is randomly chosen without context considered for workload, expense, processing time, or other metrics [45].

3.6.2. SBP Optimized Routing Efficiency [14]

The service broker regularly tracks the efficiency of all data centers in this routing policy and is dependent on the direct bottleneck to the data center with the best reaction time [16]. Virtual machines deal with customer requests in favor of maximum response and help in providing point-to-point contact much more quickly [48]. Additionally, all demands are considered to have the same processing specifications and execution times [40].

3.6.3. SBP Dynamic Reconfiguring-Based Routing [14]

On the cloud, a cloud analyst tries to exchange data center loads with several other data centers while the output of the initial data center reduces above a predetermined threshold [17]. In this scheme, the measurement is finished by taking into account these retention times, thus producing the greatest cycle time ever recorded [43] and highlighting the expense and efficiency expectations of the consumers [40]. It also corresponds with the increments and decrements of the number of virtual machines [49].

3.7. Role of Service Broker Policies

The SBP contributes to one or more (publicly or privately) cloud services on account of one or more users of that product through three main jobs: aggregation, incorporation, and configuration brokerage [22]. The cloud broker framework helps suppliers of software to collaborate with cloud vendors of organizations such as Verizon, Google, and Amazon.

Rather than recurringly billing middle customers for services offered by a cloud service, associates may provide guidance, conversion facilities, and hourly guidance, says informal platforms Vice President of Comcast Company, Craig Schlagbaum [19]. Cloud service brokers are cloud computing operations that include [11].

3.7.1. Intermediation Service

A cloud broker improves a digital service by enhancing its technical features and offering a quality-added service to cloud users, handling accessibility to cloud providers, content management, performance monitoring, improved security, and so on [11].

3.7.2. Aggregated Service

A cloud broker puts several systems together and incorporates them into one or more new services. The brokers guarantee data aggregation, and safe transfer of data across cloud users and various cloud providers [11].

3.7.3. Arbitration Service

Transaction arbitrage is equivalent to transaction consolidation in that it does not address the services becoming compiled. Service arbitrage ensures that a broker can select different agency facilities. For example, the cloud broker may use a credit-scoring service to calculate and take the best score agency [11].

A cloud broker typically provides services in three categories [11]. Figure 6 demonstrates services under these three roles.

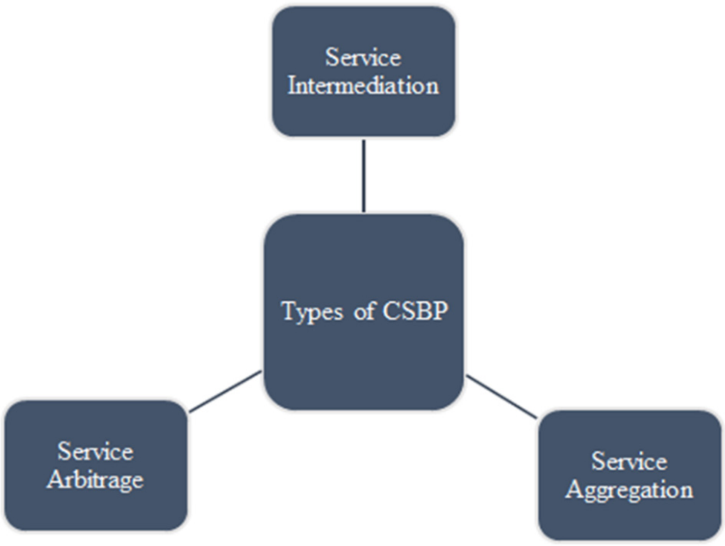


Figure 6. The cloud SBP services in three roles [11].

3.8. A Comparative Analysis of Service Broker Policies

Three quality metrics are used to examine the performance of each broker policy in Table 4 after the case study for the cloud SBP. This table gives the finest advice on how to choose the right cloud service broker for the user and deliver SLA-based cloud computing services. This comparison is used to reduce the amount of traffic between the cloud user and the cloud data center [11].

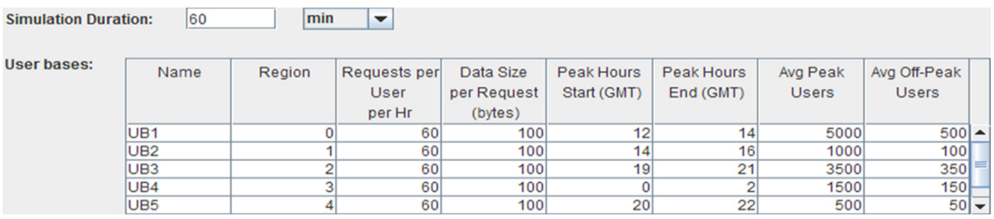
The cloud services are regulated for each user base using this approach. Furthermore, additions that focus more precisely on the effects of learning and interaction provide new avenues for research and advancement in this critical area [11].

Table 4. Comparative analysis of cloud service broker policy.

Ref.	Author Details	Service Broker Policy	Environment	Features
[1]	Ahmed I. El Karadawy et al. (2020)	Proximity-based routing	For request execution, it chooses the closest data center based on its latency.	This strategy takes advantage of the newly generated index to store information about the virtualized data centers that are now available (i.e., region–ID–statue).
[11]	Amrita Jyoti et al. (2020)		Using latency information, the closest and most cost-effective data center is chosen.	Data transport and virtual machine response time are reduced.
[2]	Sunny Nandwani et al. (2015)	Optimized routing efficiency	The fastest data centers are chosen based on their ability to handle resources.	Monitors the performance of all data centers in real time.
[5]	Divyani et al. (2018)		Feedback mechanism with a small footprint.	Increases response time and average cost performance.
[6]	Ritesh Patel and Sandip Patel (2018)	Dynamic reconfiguring-based routing	This is a supplement to the closest data center policy, including a comparative steering reason.	It increases or decreases the number of virtual machines.
[12]	Hetal and Ritesh (2015)		The number of virtual machines increases or decreases depending on the needs of the user.	Uses resources efficiently and manages different cloud services in response to user requests.

3.9. Deployment Configuration of Service Broker Policy

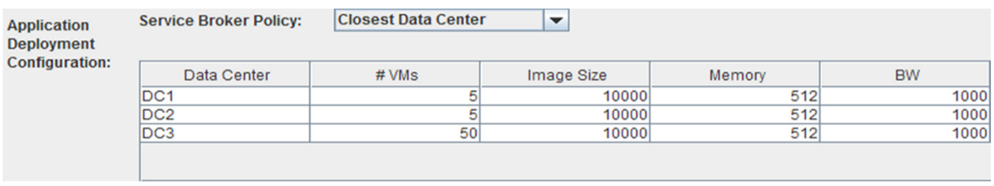
The dataset of the simulated application is represented by the configuration values for inputs. These values are based on six user bases. Figure 7 depicts the setting values for six user bases (location–peak–hours–user number). The findings show how each of the four LB algorithms (PSO, RR, throttled, and ESCE) performs with each of the three service broker policies (CDC, ORT, and reconfigure dynamically with load). The cloud analyst tool was used to analyze the various LB algorithms. To implement the various algorithms, the simulated environment operated by taking six user bases and three data centers, having five virtual machines in Data Center 1 and Data Center 2, and fifty virtual machines in Data Center 3. Each simulation was performed for 60 minutes. They considered average peak users as 500, 1500, 1000, 35,000, and 5000, and average off-peak users to be 50, 100, 150, 350, and 500 in user bases. The SBP used for simulation was CDC, which optimizes response time, and reconfigures dynamically with the load. The deployment setup is shown in Figures 8–10.


Simulation Duration: 60 min

User bases:

Name	Region	Requests per User per Hr	Data Size per Request (bytes)	Peak Hours Start (GMT)	Peak Hours End (GMT)	Avg Peak Users	Avg Off-Peak Users
UB1	0	60	100	12	14	5000	500
UB2	1	60	100	14	16	1000	100
UB3	2	60	100	19	21	3500	350
UB4	3	60	100	0	2	1500	150
UB5	4	60	100	20	22	500	50

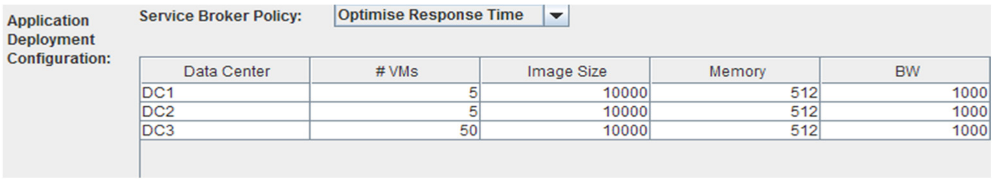
Figure 7. Six user bases with different regions and durations.


Application Deployment Configuration:

Service Broker Policy: Closest Data Center

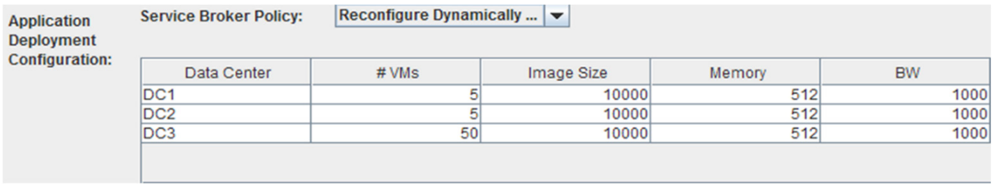
Data Center	# VMs	Image Size	Memory	BW
DC1	5	10000	512	1000
DC2	5	10000	512	1000
DC3	50	10000	512	1000

Figure 8. Application deployment configuration of Data Centers 1 to 3 based on CDC SBP.



Data Center	# VMs	Image Size	Memory	BW
DC1	5	10000	512	1000
DC2	5	10000	512	1000
DC3	50	10000	512	1000

Figure 9. Application deployment configuration of Data Centers 1 to 3 based on ORT SBP.



Data Center	# VMs	Image Size	Memory	BW
DC1	5	10000	512	1000
DC2	5	10000	512	1000
DC3	50	10000	512	1000

Figure 10. The application deployment configuration of Data Centers 1 to 3 based on reconfiguring dynamically with load SBP.

4. Results and Findings

In this section, the paper's contribution will be compared with the state-of-the-art techniques currently used in cloud analysts, with a focus on the PSO that is typically implemented by IaaS, SaaS, and PaaS cloud providers. In particular, the overall response time, DCPT, and cost of the different solutions based on service broker policies will be investigated. Moreover, RR ESCE and throttled algorithms will also be analyzed.

All of the LB algorithms for comparison are implemented into a cloud analyst simulator. A cloud analyst simulator is a tool based on CloudSim that simulates large-scale applications in a cloud environment. Cloud analyst is an extension of CloudSim with added visualization capabilities. The PSO, RR, ESCE, and throttled virtual machine LB algorithms have been configured on the cloud analyst. The selected three service broker policies are CDC, optimize response time, and reconfigure dynamically with load for experimental purposes with LB algorithms (PSO, RR, ESCE, and throttled). The following are various scenarios in which implementation has been performed.

Section 4.1 is used for the analysis of the performance parameters of PSO, RR, ESCE, and throttled algorithms exploiting the CDC, optimizing response time, and reconfiguring dynamically with load service broker policies to capture a more realistic scenario where performance may change due to overall response time, DCPT, and cost parameters.

4.1. Analysis of Load-Balancing Algorithms

This scenario is created to analyze PSO, RR, ESCE, and throttled algorithms' performance parameters based on the CDC, optimize response time, and reconfigure dynamically with load service broker policies, and the goal of this scenario is to figure out overall response time, DCPT, and cost parameters (see Figures 11–14). Furthermore, the simulation environment consists of six user bases and three data centers, having five virtual machines in Data Center 1 and Data Center 2 and fifty virtual machines in Data Center 3, and each simulation is performed for 60 minutes with the average peak users being 500, 1500, 1000, 35,000, and 5000 and average off-peak users being 50, 100, 150, 350, and 500 in user bases.

4.2. Comparison of Load-Balancing Algorithms' Results

To analyze and compare the results of LB algorithms obtained by executing the above set of algorithms, the results compare PSO, RR, ESCE, and throttled LB algorithms by considering the overall response time, DCPT, and cost parameters for each of the three service broker policies, which are CDC, optimize response time, and reconfigure dynamically with the load.

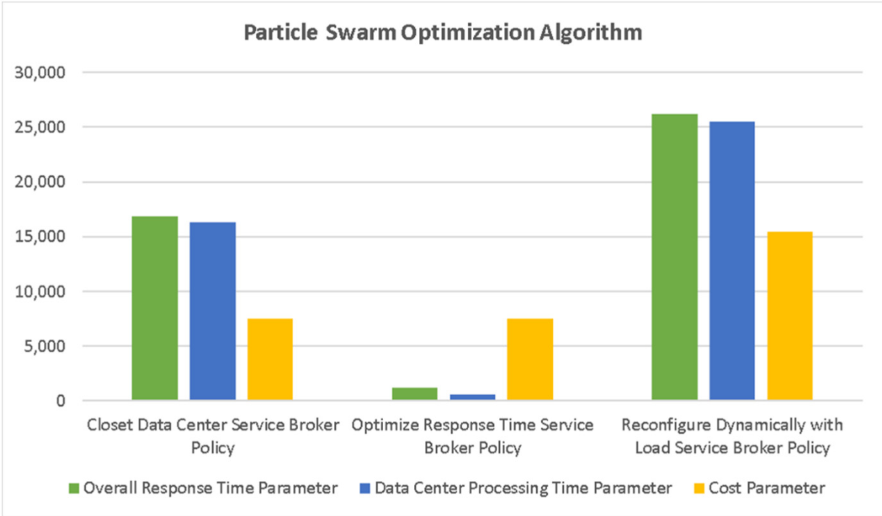


Figure 11. The analysis of the performance parameters of PSO based on 3 service broker policies.

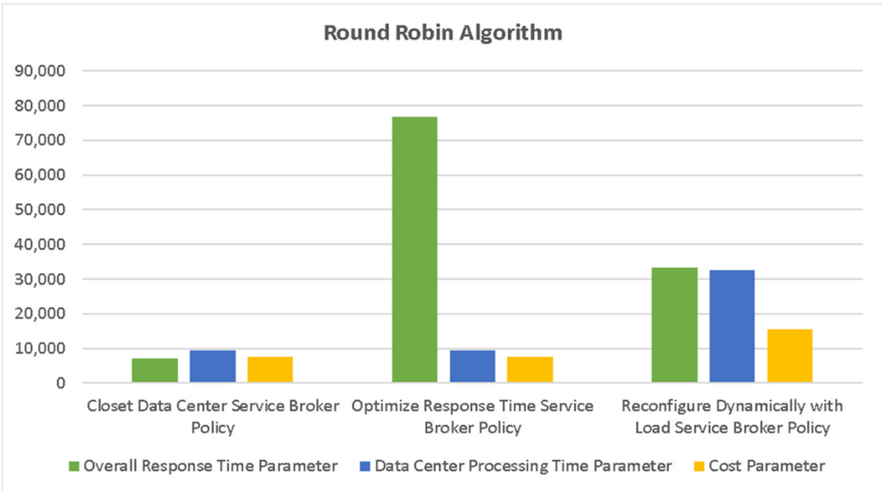


Figure 12. The analysis of the performance parameters of RR based on 3 service broker policies.

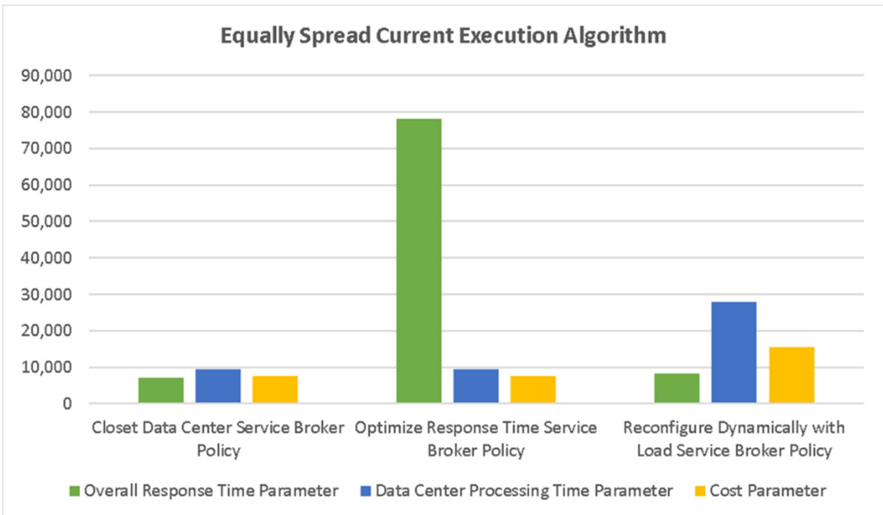


Figure 13. The analysis of the performance parameters of the ESCE algorithm on 3 service broker policies.

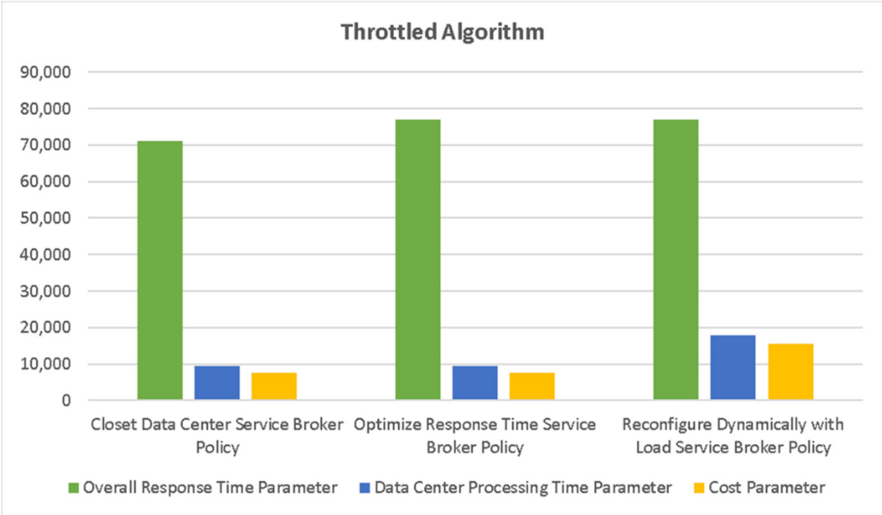


Figure 14. The analysis of the performance parameters of the throttled algorithm on 3 service broker policies.

5. Discussion

The findings compare the PSO, RR, ESCE, and throttled LB algorithms by taking into account overall response time, DCPT, and cost parameters for each of the CDC, optimizing response time, and reconfiguring dynamically with load service broker policies.

5.1. Closest Data Center Service Broker Policy

The observation in this CDC SBP is that the two LB algorithms RR and ESCE perform almost the same for overall response time, DCPT, and cost parameters, but throttled and PSO algorithms performed higher than RR and ESCE LB algorithms. However, the values of DCPT and cost parameters are almost the same for PSO, RR, ESCE, and throttled. This is because all requests are passed to the closest data center SBP. Figure 15 shows the output of this observation.

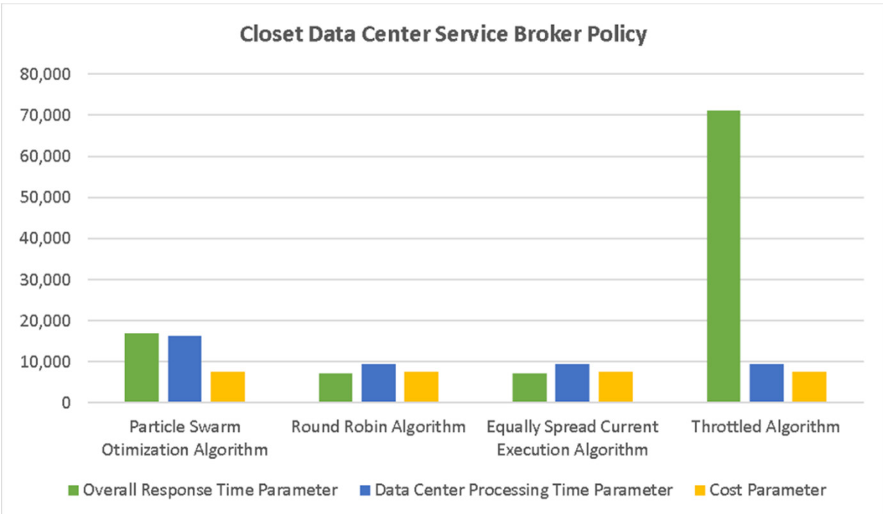


Figure 15. The comparison of PSO and RR, ESCE, and throttled algorithms by considering the parameters based on CDC SBP.

5.2. Optimize Response Time Service Broker Policy

The observation in this ORT SBP is that the three LB algorithms RR, ESCE, and throttled are performed almost the same for overall response time, DCPT, and cost parameters but

PSO algorithms have lower output values than them. This is because the ORT SBP delivers the majority of requests to the closest data center SBP and selects the data center with the shortest response time for the remaining requests. Figure 16 shows the output of this observation.

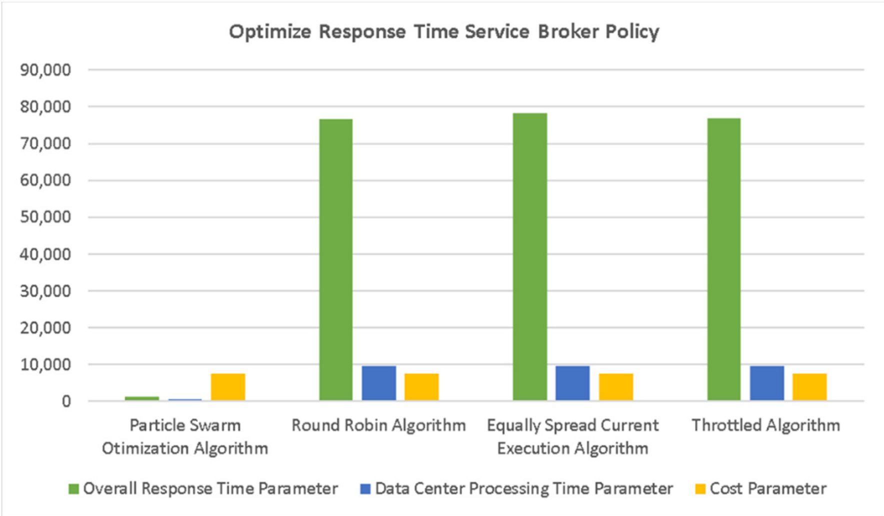


Figure 16. The comparison of PSO, RR, ESCE, and throttled algorithms by considering the parameters based on the optimizing response time SBP.

5.3. Reconfigure Dynamically with Load Service Broker Policy

The observations show with regard to this reconfigure dynamically with load SBP that the four LB algorithms PSO, RR, ESCE, and throttled performed almost the same for overall response time, DCPT, and cost parameters but the throttled algorithm has higher output values than them. The overhead of scaling the virtual machines up or down depending on the peak load is added to the reconfiguring dynamically with load SBP. As a result, the numbers obtained in this scenario for overall response time and DCPT are greater than the cost parameter. Figure 17 shows the output of this observation.

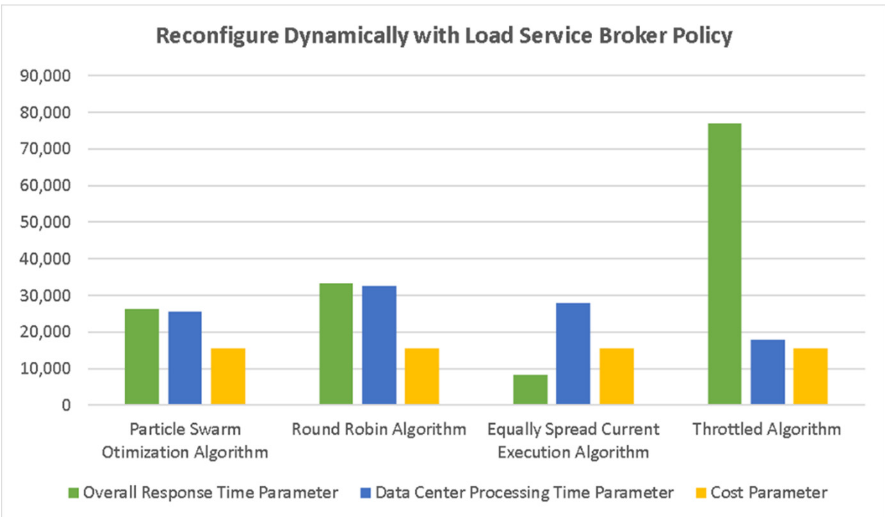


Figure 17. The comparison of PSO, RR, ESCE, and throttled algorithms by considering the parameters based on the reconfiguring dynamically with load SBP.

6. Conclusions and Future Work

Cloud computing technology is rapidly advancing around the world, necessitating the development of effective LB strategies to deliver better cloud services to consumers and adapt to the rapid technological advancements of the cloud environment. The factual analysis included performance parameters which were overall response time, DCPT, and cost of LB algorithms, which were PSO, RR, ESCE, and throttled, as well as service broker policies which were CDC, optimize response time, and reconfigure dynamically with load; these were all studied in this paper. For each algorithm, we provided the results. In this study, PSO, RR, ESCE, and throttled algorithms have been discussed. These algorithms can improve overall response time and DCPT and minimize the execution cost of cloud applications guaranteeing the respect of multi-class SLAs. A cloud analyst simulator is a tool based on CloudSim that simulates large-scale applications in a cloud environment. Cloud analyst is an extension of CloudSim with added visualization capabilities.

In future work, the LB algorithms can be extended to develop new innovative service broker policies capable of integrating technologies such as machine learning and artificial intelligence to respond to the circumstances dynamically. Another area of research lies in expanding the cloud analyst tool functionality to fit new algorithms and develop the interface. The enhancement of the simulation module throughout the graphics simulation also has scope.

Author Contributions: Conceptualization, M.A.S.; methodology, M.A.S. and M.M.A.; validation, M.A.S. and M.M.A.; investigation, M.A.S. and M.M.S.; resources, M.A.S. and M.M.A.; writing—original draft preparation, M.A.S., M.M.A.; writing—review and editing, M.A.S., M.M.A., and M.M.S.; visualization, M.A.S.; supervision, M.A.S. and M.M.A. All authors have read and agreed to the published version of the manuscript.

Funding: The research will be funded by the Multimedia University, Persiaran Multimedia, 63100 Cyberjaya, Selangor, Malaysia.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. El Karadawy, A.I.; Mawgoud, A.A.; Rady, H.M. An Empirical Analysis on Load Balancing and Service Broker Techniques using Cloud Analyst Simulator. In Proceedings of the 2020 IEEE International Conference on Innovative Trends in Communication and Computer Engineering (ITCE), Aswan, Egypt, 8–9 February 2020; pp. 27–32.
2. Nandwani, S.; Achhra, M.; Shah, R.; Tamrakar, A.; Joshi, K.; Raksha, S. Analysis of service broker and load balancing in cloud computing. *Int. J. Curr. Eng. Sci. Res. (IJCESR)* **2015**, *2*, 92–98.
3. Suguna, D.; Barani, R. Simulation of Dynamic Load Balancing Algorithms. *Bonfring Int. J. Softw. Eng. Soft Comput.* **2015**, *5*, 1–7. [CrossRef]
4. Singh, S.P.; Sharma, A.; Kumar, R. Analysis of Load Balancing Algorithms using Cloud Analyst. *Int. J. Grid Distrib. Comput.* **2016**, *9*, 11–24. [CrossRef]
5. Kumar, D.R.; Bhattacharya, S. Performance evaluation of Load Balancing with Service Broker policies for various workloads in cloud computing. *Int. J. Res. Trends Innov.* **2018**, *3*, 136–142.
6. Patel, R.; Patel, S. Efficient Service Broker Policy for Intra Datacenter Load Balancing. In *Information and Communication Technology for Intelligent Systems*; Satapathy, S.C., Joshi, A., Eds.; Springer: Singapore, 2019; pp. 683–692.
7. Manasrah, A.M.; Smadi, T.; ALmomani, A. A Variable Service Broker Routing Policy for data center selection in cloud analyst. *J. King Saud Univ.-Comput. Inf. Sci.* **2017**, *29*, 365–377. [CrossRef]
8. Singh, M.; Nandal, P.; Bura, D. Comparative Analysis of Different Load Balancing Algorithm Using Cloud Analyst. In *Data Science and Analytics*; Panda, B., Sharma, S., Roy, N.R., Eds.; Springer: Singapore, 2018; pp. 321–329.
9. Shakir, M.S.; Razzaque, E.A. Performance Comparison of Load Balancing Algorithms using Cloud Analyst in Cloud Computing. In Proceedings of the 2017 IEEE 8th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON), New York, NY, USA, 19–21 October 2017; Volume 5.

10. Jain, A.; Kumar, R. A multi stage load balancing technique for cloud environment. In Proceedings of the 2016 International Conference on Information Communication and Embedded Systems (ICICES), Chenai, India, 25–26 February 2016; pp. 1–7.
11. Jyoti, A.; Shrimali, M.; Tiwari, S.; Singh, H.P. Cloud computing using load balancing and service broker policy for IT service: A taxonomy and survey. *J Ambient Intell Hum. Comput* **2020**, *11*, 4785–4814. [CrossRef]
12. Patel, H.V.; Patel, R. Cloud Analyst: An Insight of Service Broker Policy. *Int. J. Adv. Res. Comput. Commun. Eng.* **2015**, *4*, 122–127. [CrossRef]
13. Benlalia, Z.; Beni-hssane, A.; Abouelmehdi, K.; Ezati, A. A new service broker algorithm optimizing the cost and response time for cloud computing. *Procedia Comput. Sci.* **2019**, *151*, 992–997. [CrossRef]
14. Radi, M. Efficient Service Broker Policy For Large-Scale Cloud Environments. *Int. J. Comput. Sci.* **2015**, *12*, 85–90.
15. Chugh, S.; Verma, A.; Sharma, K.; Tech, M. Data Center Processing Time Evaluation of Service Broker Policies In A single Data Center. *Int. J. Eng. Res. Gen. Sci.* **2015**, *3*, 325–329.
16. Mesbahi, M.R.; Hashemi, M.; Rahmani, A.M. Performance evaluation and analysis of load balancing algorithms in cloud computing environments. In Proceedings of the 2016 Second International Conference on Web Research (ICWR), Tehran, Iran, 27–28 April 2016; pp. 145–151.
17. Nema, R.; Edwin, S.T. A Review: Load balancing Algorithm Using Cloud Analyst Environment. *Int. J. Sci. Manag. Eng. Res. IJSMER* **2016**, *1*, 192–197.
18. Definition of Cloud Services Brokerage (CSB). Gartner Information Technology Glossary. In Gartner. Available online: https://www.gartner.com/en/information-technology/glossary/cloud-services-brokerage-csb (accessed on 29 June 2021).
19. Patel, S.; Patel, R.; Patel, H.; Vahora, S. CloudAnalyst: A Survey of Load Balancing Policies. *Int. J. Comput. Appl.* **2015**, *117*, 21–24. [CrossRef]
20. Available online: https://shodhganga.inflibnet.ac.in/bitstream/10603/169324/15/15_chapter3.pdf (accessed on 28 December 2020).
21. Joshi, S.; Kumari, U. Load balancing in cloud computing: Challenges & issues. In Proceedings of the 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I), Greater Noida, India, 14–17 December 2016; pp. 120–125.
22. Kaneria, O.; Banyal, R.K. Analysis and improvement of load balancing in Cloud Computing. In Proceedings of the 2016 International Conference on ICT in Business Industry & Government (ICTBIG), Indore, India, 18–19 November 2016; pp. 1–5.
23. Yagnesh, R.P.; Thaker, C. A Comparative Survey on Various Load Balancing Techniques in Cloud Computing. *IJSRSET Int. J. Sci. Res. Sci. Eng. Technol.* **2015**, *1*, 250–255.
24. Kumar, A.; Kalra, M. Load balancing in cloud data center using modified active monitoring load balancer. In Proceedings of the 2016 International Conference on Advances in Computing, Communication, & Automation (ICACCA) (Spring), Dehradun, India, 8–9 April 2016; pp. 1–5.
25. Panwar, R.; Mallick, B. Load balancing in cloud computing using dynamic load management algorithm. In Proceedings of the 2015 International Conference on Green Computing and Internet of Things (ICGCIoT), Greater Noida, India, 8–10 October 2015; pp. 773–778.
26. Farid, M.; Latip, R.; Hussin, M.; Abdul Hamid, N.A.W. A Survey on QoS Requirements Based on Particle Swarm Optimization Scheduling Techniques for Workflow Scheduling in Cloud Computing. *Symmetry* **2020**, *12*, 551. [CrossRef]
27. Kaur, K.; Mahajan, R. Equally Spread Current Execution Load Algorithm—A Novel Approach for Improving Data Centre's Performance in Cloud Computing. *Int. J. Future Revolut. Comput. Sci. Commun. Eng.* **2018**, *4*, 8–10.
28. Ettikyala, K.; Rama Devi, Y. A Study on Cloud Simulation Tools. *Int. J. Comput. Appl.* **2015**, *115*, 18–21. [CrossRef]
29. Sinha, U.; Shekhar, M. Comparison of Various Cloud Simulation tools available in Cloud Computing. *Int. J. Adv. Res. Comput. Commun. Eng.* **2015**, *4*, 171–176. [CrossRef]
30. Jyoti, A.; Shrimali, M. Dynamic provisioning of resources based on load balancing and service broker policy in cloud computing. *Clust. Comput* **2020**, *23*, 377–395. [CrossRef]
31. Pant, M.; Sharma, T.K.; Verma, O.P.; Singla, R.; Sikander, A. *Soft Computing: Theories and Applications: Proceedings of SoCTA 2018*; Springer: Singapore, 2020.
32. Singh, A.; Kumar, R. Performance Evaluation of Load Balancing Algorithms Using Cloud Analyst. In Proceedings of the 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 29–31 January 2020; pp. 156–162.
33. Sharma, H.; Govindan, K.; Poonia, R.C.; Kumar, S.; Wael, M. *Advances in Computing and Intelligent Systems: Proceedings of ICACM 2019*; Springer: Singapore, 2020.
34. Al-Tarawneh, M.; Al-Mousa, A. Adaptive user-oriented fuzzy-based service broker for cloud services. *J. King Saud Univ.-Comput. Inf. Sci.* **2019**, *34*, 354–364. [CrossRef]
35. Valarmathi, R.; Sheela, T. Differed service broker scheduling for data centres in cloud environment. *Comput. Commun.* **2019**, *146*, 186–191. [CrossRef]
36. Arseniev, D.G.; Overmeyer, L.; Kälviäinen, H.; Katalinič, B. *Cyber-Physical Systems and Control*; Springer International Publishing: St. Petersburg, Russia, 2020.
37. Challa, R.K.; Dutta, M.; Kumar, R. *Proceedings of 2nd International Conference on Communication, Computing and Networking: ICCCN 2018, NITTTR Chandigarh, India*; Springer: Singapore, 2019.
38. Raghuwanshi, S.; Kapoor, S. The New Service Brokering Policy for Cloud Computing Based on Optimization Techniques. *Int. J. Eng. Tech.* **2018**, *4*, 481–488.

39. Nema, M.R. A new efficient Virtual Machine load balancing Algorithm for a cloud computing environment. *Int. J. Latest Res. Eng. Technol. (IJLRET)* **2016**, *2*, 69–75.
40. Nayak, S.; Patel, P. Analytical Study for Throttled and Proposed Throttled Algorithm of Load Balancing in Cloud Computing using Cloud Analyst. *Int. J. Sci. Technol. Eng.* **2015**, *1*, 90–100.
41. Mahalle, H.S.; Tayde, S.; Kaveri, P.R. Implementing Service Broker Policies in Cloud Computing Enviroment. In Proceedings of the 2015 International Conference on Communication Networks (ICCN), Gwalior, India, 19–21 November 2015.
42. Gaur, A.; Garg, K. Survey paper on cloud computing with load balancing policy. *Int. J. Eng. Res.* **2015**, *2*, 7.
43. Khodar, A.; Mager, V.E.; Alkhayat, I.; Al-Soudani, F.A.; Desyatirikova, E.N. Evaluation and Analysis of Service Broker Algorithms in Cloud-Analyst. In Proceedings of the 2020 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), St. Petersburg, Russia, 27–30 January 2020; pp. 351–355.
44. Jena, S.R.; Shanmugam, R.; Saini, K.; Kumar, S. Cloud Computing Tools: Inside Views and Analysis. *Procedia Comput. Sci.* **2020**, *173*, 382–391. [CrossRef]
45. Parida, S.; Pati, B. A Cost Efficient Service Broker Policy for Data Center Allocation in IaaS Cloud Model. *Wirel. Pers Commun* **2020**, *115*, 267–289. [CrossRef]
46. Arulkumar, V.; Bhalaji, N. Performance analysis of nature inspired load balancing algorithm in cloud environment. *J Ambient Intell Hum. Comput.* **2021**, *12*, 3735–3742. [CrossRef]
47. The Best Open Source Cloud Computing Simulators. Available online: https://www.opensourceforu.com/2016/11/best-open-source-cloud-computing-simulators/ (accessed on 11 June 2021).
48. Khalil, K.M.; Abdel-Aziz, M.; Nazmy, T.T.; Salem, A.-B.M. Cloud Simulators—An Evaluation Study. *Int. J. Inf. Model. Anal.* **2017**, *6*, 3–25.
49. Bahwaireth, K.; Tawalbeh, L.A.; Benkhelifa, E.; Jararweh, Y.; Tawalbeh, M.A. Experimental comparison of simulation tools for efficient cloud and mobile cloud computing applications. *EURASIP J Inf Secur.* **2016**, *2016*, 15. [CrossRef]
50. Suryateja, P.S. A Comparative Analysis of Cloud Simulators. *Int. J. Mod. Educ. Comput. Sci.* **2016**, *8*, 64–71. [CrossRef]
51. Fakhfakh, F.; Kacem, H.H.; Kacem, A.H. Simulation tools for cloud computing: A survey and comparative study. In Proceedings of the 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), Wuhan, China, 24–26 May 2017; pp. 221–226.
52. Shahid, M.A.; Islam, N.; Alam, M.M.; Su'ud, M.M.; Musa, S. A Comprehensive Study of Load Balancing Approaches in the Cloud Computing Environment and a Novel Fault Tolerance Approach. *IEEE Access* **2020**, *8*, 130500–130526. [CrossRef]
53. Shahid, M.A.; Islam, N.; Alam, M.M.; Mazliham, M.S.; Musa, S. Towards Resilient Method: An exhaustive survey of fault tolerance methods in the cloud computing environment. *Comput. Sci. Rev.* **2021**, *40*, 100398. [CrossRef]