Journal of Data Acquisition and Processing
ISSN: 1004-9037
https://sjcjycl.cn/
DOI: 10.5281/zenodo.778202

ANALYSIS AND PERFORMANCE ASSESSMENT OF CLOUD COMPUTING USING LOAD BALANCING ALGORITHMS

[1]Devendra Namdeo, [2]Dr. Jyotibala Gupta, [3]Dr. Praveen Kumar Shrivastava
[1]Research Scholar, Dr. C.V.Raman University, Kota, Bilaspur (C.G.) India
[2]Asst. Professor, Dr. C.V.Raman University, Kota, Bilaspur (C.G.) India
[3]Dept, of Information Technology, K.I.T. Raigarh (C.G.) India

Abstract-
One of the major issues in the field of cloud computing today is distributing the system burden and distributing all incoming requests among all processing nodes. Numerous load balancing techniques and algorithms have been put forth for cloud and distributed computing systems. Furthermore, one of the key elements for enhancing system performance in a cloud context is the broker policy for allocating the workload among several datacenters. An analytical comparison between various broker policies and VM load balancing techniques is presented in this study. We simulate these methods using the Cloud Analyst simulator to assess them, and the final results are shown depending on various factors. The optimal combinations are indicated by the research's findings.

***Keyword: -* Cloud Computing, *Virtual Machines, Load Balancing, Broker Policy, Performance Evaluation*.**

I. INTRODUCTION

Platforms for cloud computing are becoming more and more prevalent these days. Cloud computing, sometimes known as just "the cloud," is the pay-per-use online delivery of on-demand computer resources. According to the official definition provided by NIST, "cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction". In general, everything that includes providing hosted services over the Internet is referred to as "cloud computing."

Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS), and Software-as-a-Service (SaaS) are the three primary categories into which these services fall. The majority of IT departments are compelled to dedicate a large amount of their time on tedious maintenance, upgrade, and implementation initiatives. These days, however, IT departments are using cloud computing technology to cut down on time spent on low-value tasks and free up resources for more strategically important IT initiatives that have a bigger business impact. There are three primary features that set cloud computing services apart from traditional hosting. It is offered for sale on demand, typically hourly or minutely; it offers elasticity property, meaning a customer can have as much or as little of a service as they require at any one moment, and at the end, the cloud service providers handle all aspect of the services management. In addition to all of the benefits of cloud computing, there are a number of obstacles and unresolved problems in the field of cloud computing research, including issues with load balancing, energy efficiency and green computing, job scheduling and security.

One of the key concepts in cloud computing environments and distributed systems in general is load balancing, which influences system performance based on the quantity of work assigned to the system for a predetermined amount of time. Redistributing the overall system burden among system resources in order to maximize resource consumption and boost system performance is known as load balancing. Users will be satisfied by boosting throughput and reducing response times because load balancing has been taken into account to ensure that each virtual machine in the cloud computing system performs the same amount of duty.

With our method, we offer an analytical comparison and performance assessment of all popular load balancing algorithms that are suggested and simulated in the cloud computing simulator CloudAnalyst. It makes it possible for customers to quickly and simply assess the regional distribution requirements of large-scale Cloud applications.

We assess every potential configuration of load balancing techniques in each datacenter and datacenter broker strategy for allocating incoming workloads across available datacenters under the same exhaustive scenario. By simulating all of these various scenarios, we will provide the optimal mix of these policies and load balancing techniques for an analytical comparison.

The following topics are covered in the remaining portions of this paper: Some related works are reviewed in Section II. We will look over the suggested scenario in part III, along with some fundamental ideas on the primary datacenter broker policies and load balancing algorithms that are suggested on CloudAnalyst. The simulation results are displayed in Section IV, along with an examination of various combinations of datacenter broker policies and load balancing algorithms based on the simulation results. Section V will finally include our proposal for this paper's conclusion and next steps.

II. RELATED WORKS

We will discuss a few of the research that have been done on load balancing performance evaluation and that include various load balancing methods in cloud computing environments in this part. A comparative analysis of two virtual machine load balancing algorithms—Round Robin and Throttled has been presented in. The present investigation employs Round Robin and Throttled virtual machine load balancing policies in conjunction with an optimized response time service broker policy. The simulation is conducted through parameter adjustments to examine various performance-related factors, including overall response time, datacenter hourly average processing times, datacenter request servicing times, response times based on region, user base hourly response times, and total cost. The simulation findings show that in a heterogeneous cloud computing environment, the combination of the suggested approach of throttled and optimized response time service broker policy performs better than the round robin load balancing algorithm.

An examination of several cloud computing load balancing algorithms in order to determine qualitative elements for simulation and assess the load balancing algorithms' execution times. Three load balancing methods have been used in this study's simulation process: Round Robin, Central Queuing, and Randomized with different combinations of MIPS vs. Host and MIPS vs. VM. The results of the simulation indicate that response time is inversely proportional to MIPS vs. VM and MIPS vs. Host; nevertheless, the ideal response time is attained when MIPS vs. VM and MIPS vs. Host have the same value.

Three distributed load balancing techniques have been compared for cloud computing scenarios in. Three typical algorithms were selected for performance comparison in this study. The first was based directly on a naturally occurring phenomenon: honey bee foraging; the second employed biased random sampling to try and create a desired global outcome; and the third involved rewiring the system in a process known as active clustering. The simulation findings show that when a variety of service kinds are needed, the honeybee-based load balancing methods perform better. Furthermore, the simulation demonstrates that random sampling walks decline rapidly with increased population diversity and perform better when verifying similar groups. Similar to a random walk, Active Clustering performs better as the number of processing nodes increases.

The authors compared the performance of several virtual machine load balancing algorithms and cloud computing regulations. Four popular load balancing algorithms have been taken into consideration in this study. First Come First Serve, Execution Load, Round Robin, and Throttled Load Balancing Algorithms have all been examined in light of the overall cost, average response time, and average datacenter request servicing time. The CloudAnalyst simulator's simulation findings indicate that round robin offers the best integration performance.

III. LOAD BALANCING ALGORITHMS AND POLICIES, AGGREGATED SCENARIO

We looked at a few related load balancing performance evaluation studies in cloud computing that suggested simulating virtual machine load balancers in the preceding section. However, all of the earlier studies only addressed load balancing in cloud datacenters; in contrast, the approach of allocating the workload among cloud datacenters—which is often handled by datacenter brokers—is incredibly successful at balancing loads and simulating outcomes. We will examine the cloud computing load balancing process at two different stages in our methodology.

The initial level of the CloudAnalyst simulator presents a model of service brokers that manages traffic routing between user bases and datacenters. This model is called CloudAppServiceBroker. The CloudAnalyst simulator has three standard and default routing policies: "Reconfigure Dynamic with Load," "Optimize Response Time," and "Closest Datacenter." The second level, which the VMLoadBalancer component introduces into CloudAnalyst, is in charge of simulating the load balance strategy that datacenters employ to serve allocation requests. The simulator provides three standard load balancing techniques in each datacenter: "Equally Speared Current Execution Load," "Round Robin," and "Throttled." Nine distinct results are obtained by varying the combinations of these three VM load balancing algorithms and datacenter broker Policies. These results will be examined in the remaining sections of this study using various evaluation criteria, including total response time, datacenter processing time, and cost. The simulated situation, virtual machine load balancing methods, and datacenter broker policies will all be covered in more detail in the sections that follow.

a. Simulated Scenario

The CloudAnalyst simulator's simulated situation is depicted in Figure 1. For every possible combination of load balancing techniques, we simulate under the same conditions using the same scenario. As depicted in Fig. 1, the simulated scenario includes two datacenters and three users, each of whom is positioned on the map in a distinct geographic area. There is no user base and datacenter 1 in region 0. In this region, R1 has just one user and no datacenter, whereas R5 has one user and no datacenter, and R4 has one datacenter and one user base. We attempted to cover every eventuality that could arise throughout the simulation process with this type of scenario setting.

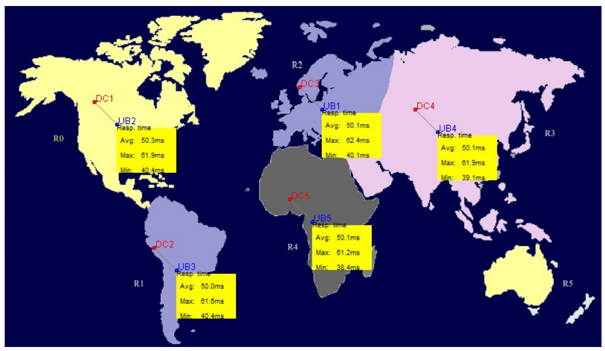


Fig. 1. The CloudAnalyst scenario on map

b. Datacenter Brocker Policies

The traffic routing between user bases and datacenters is managed by service broker policies. On the CloudAnalyst simulator, three distinct datacenter broker policies have been put into place.Traffic from the originating user base is routed to the closest datacenter in terms of network latency by the default routing policy, known as the "Closest Data Center" policy (ClosestP). The second policy, known as the "Optimize Response Time" policy (OptP), directs initial traffic to the network latency point that is closest to the source of the requests. The service broker then looks for the service broker with the best response time at that moment and divides the load across the closest and fastest data centers if the nearest data center's response time starts to deteriorate. When a datacenter's performance falls below a predetermined level, the third load sharing mechanism—known as "Reconfigure Dynamically with Load" policy (ReconfigP) on CloudAnalyst tries to spread the load between the datacenter and other datacenters.

c. VM Load Balancing Algorithms

Datacenters employ virtual machine load balancing techniques to serve allocation requests and balance the overall workload in the datacenter. The three "Round Robin," "Throttled," and "Equally Spread Current Execution Load" VM load balancing techniques that have been suggested in the literature are implemented on the CloudAnalyst simulator. We provide and briefly discuss the general characteristics of various load balancing techniques in this section.

- **Round Robin (RR)**

Round robin is one of the most popular and straightforward scheduling and load balancing algorithms that makes use of the time slice principle. Round robin is the default load balancing technique on CloudAnalyst; it distributes all incoming requests to the available virtual machines in a round robin manner without taking into account each virtual machine's current load. This policy is not regarded as planned scheduling policy with priority. Round robin architecture has a downside in that large response times deteriorate system performance.

- **Throttled**

When clients send requests to the load balancer, the throttled algorithm starts by allocating an appropriate virtual machine. Each virtual machine's processing of requests is throttled to a certain point using this VM load balancing algorithm. The primary responsibility of a throttled load balancer is to maintain an index table including all virtual machines and their respective busy and available modes. The load balancer returns a value of -1 and the datacenter queues the request until a virtual machine becomes available if client requests cause this threshold to be exceeded in all virtual machines that are available.

- **Equally Spread Current Execution (ESCE)**

The jobs are distributed across the available virtual machines (VMs) using an equally spread current execution mechanism, which evens out the quantity of active tasks on each VM at any one time. The system burden is prioritized via the ESCE algorithm. The datacenter burden is randomly distributed by ESCE, which verifies the size and transfers the load to the virtual machine with the lowest load.

This approach determines which virtual machine has the fewest allocations while maintaining an even distribution of active workloads across all VMs.

We will show the simulation results of using these datacenter broker policies and VM load balancing methods in conjunction in the following section. Our method differs from literature review studies primarily in that it simulates under a comprehensive and distinct scenario and offers a thorough analytical comparison of multiple outcome factors.

IV. A COMPARISON BETWEEN ANALYTICAL AND EXPERIMENTAL RESULTS

As previously indicated, we ran simulations using various combinations of virtual machine load balancers and datacenter broker policies in the same scenario, which included two datacenters, three user bases, and four distinct geographic locations. Every datacenter has three physical servers and uses a time-shared mechanism to divide the resources among its virtual machines. For every iteration, we run the simulation for roughly 60 minutes.

TABLE -1. SIMULATION CONFIGURATION SUMMARY

S.NO.	CLOUD RESOURCES	NUMBER OF PROCESSOR PER EACH PHYSICAL SERVER	COST PER VM ($/HR.)	OS / ARCH	VMM	DATA TRANSFER COST ($/GB)
1	Datacenter-1 (Area 0)	4	0.1	Linux / X86	Xen	0.1
2	Datacenter-2 (Area 4)	4	0.1	Linux / X86	Xen	0.1

In the same circumstance, we simulate nine different load balancing strategies. In five steps, we raise the cloudlet lengths from 100 to 5000 bytes, simulating 45 distinct simulations iterates in the process. Table I provides a full view of the simulation process. In the remaining portions of this section, we will analyze the simulation findings. Datacenter broker policies use the delay latency matrix, which is displayed in Figure 2, to choose the destination datacenter.

Region\Region	0	1	2	3	4	5
0	25	100	150	250	250	100
1	100	25	250	500	350	200
2	150	250	25	150	150	200
3	250	500	150	25	500	500
4	250	350	150	500	25	500
5	100	200	200	500	500	25

Fig. 2. CloudAnalyst delay matrix configuration

A. Closet Data Center Policy (ClosestP)

We chose ClosestP as the datacenter broker policy and used three different VM load balancing algorithms—RR, Throttled, and ESCE—to simulate the identical workload. The average response time for all datacenters and user bases is shown in Figure 3.

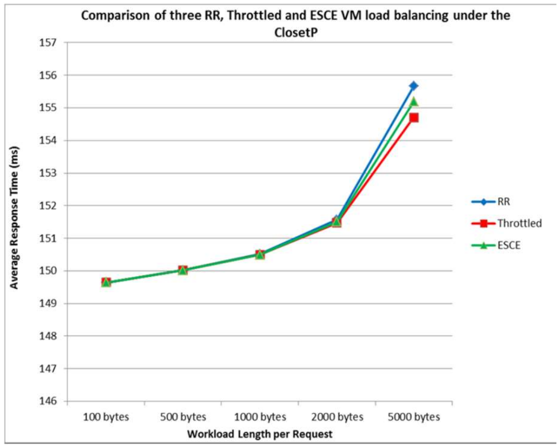


Fig. 3. Comparison of three RR, Throttled and ESCE VM load balancing under the ClosetP

As the volume per request of the datacenter workload increases, as illustrated in Fig. 3, the Throttled load balancing method performs better than the others when combined with the Closets datacenter policy. By dispersing the task randomly, the likelihood of under loaded and overloaded virtual machines will rise as the workload increases. Because the RR method distributes the load among system nodes without taking into account their existing loads, it doesn't perform as optimally in this scenario as the results indicate. However, by employing the throttled threshold and not allowing job requests to be sent to VMs that have work to complete, the throttled algorithm maintains the usual state of demand on all virtual machines.

As a result, the system performance won't suffer when the Throttled method is used, and the average response time will be better when a lot of requests are coming in. The ESCE method divides the future work load among the virtual machines (VMs) based on the number of tasks assigned to each VM; it does not take the length of the workload into account. Then, it performs better than RR but falls short of a throttled load balancer because it is insensitive to the workload of virtual machines.

B. Optimize Response Time Policy (OptP)

The OptP has been selected as the datacenter broker policy, and the simulation procedure was run using the same workload and three different VM load balancing algorithms: RR, Throttled, and ESCE. Similar to earlier policies, this broker policy uses the delay matrix at the first level to determine the destination datacenter. The average response time for all datacenters and user bases is displayed in Figure 4.

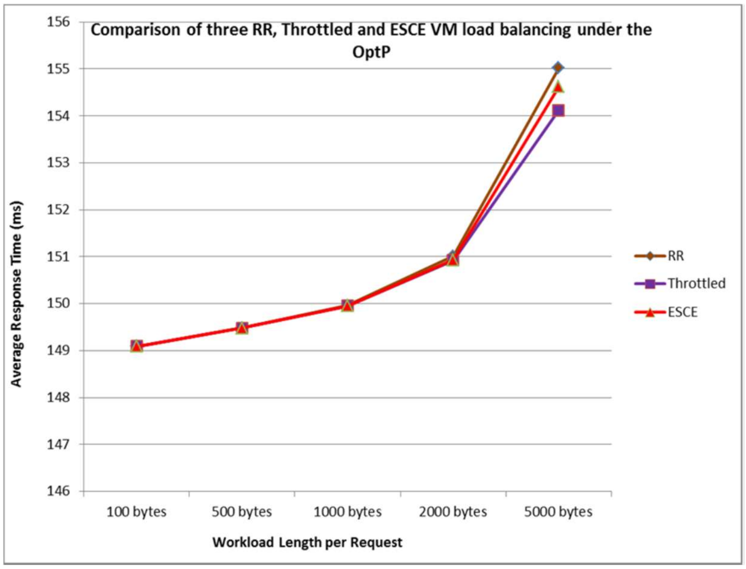


Fig. 4. Comparison of three RR, Throttled and ESCE VM load balancing under the OptP

The simulation findings in this instance are comparable to the earlier scenario, as Fig. 4 illustrates. Once more, the throttling technique outperforms other VM load balancing algorithms in terms of overall average response time since it ensures that, even with a rise in incoming requests, system performance remains stable and that the demands assigned to this datacenter will be fulfilled by available virtual machines.

According to the simulation results, the RR algorithm performs better in this scenario than the previous one because, under this datacenter broker policy, the closet datacenter will receive the initial traffic. However, if response time starts to decline, the broker policy will split the load between the closet and the fastest datacenter. Consequently, the round robin approach will perform better in this scenario by limiting the number of overloaded virtual machines.

C. Reconfigure Dynamically with Load Policy (ReconfigP)

For 15 cycles, we replicated the combination of three VM load balancing methods with ReconfigP, just as in earlier scenarios. The simulation findings, which included some surprising variances, are displayed in Figure 5.

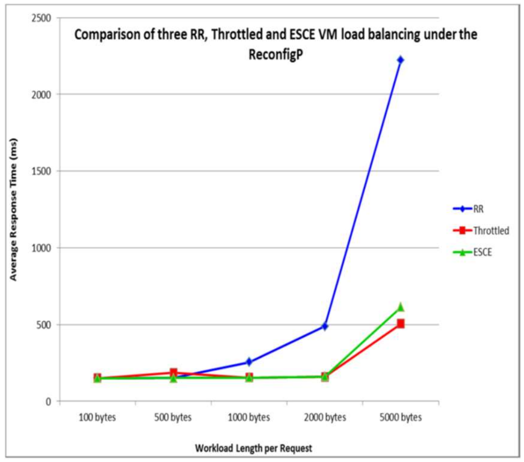


Fig. 5. Comparison of three RR, Throttled and ESCE VM load balancing under the ReconfigP

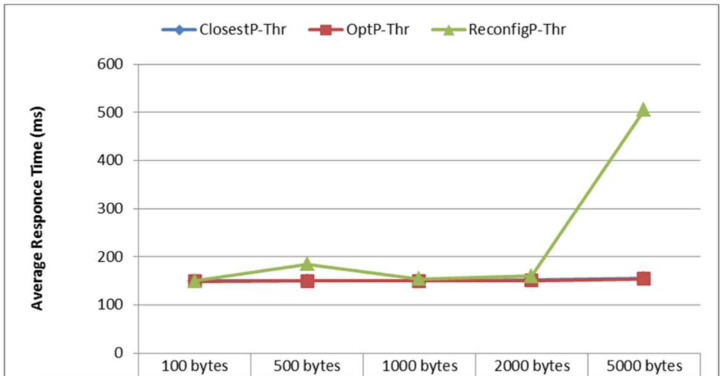
When comparing the outcomes of this instance to those of two earlier examples, they differ significantly. Although there is still a noticeable difference in speed between a 500-byte job and a 1000-byte workload, the Throttled method still performs better. When the workload length approaches 500 bytes, a significant amount of work is imposed on the datacenter, leading to a performance degradation. This is because the dynamic reconfiguration policy is unable to provide a suitable configuration and share the load of one datacenter with another. The combination of RR and ReconfigP has the worst result for this VM load balancing algorithm because the increasing amount of system workload and balancing it by RR without considering the current VMs' load caused to heavy overloaded situation. In addition, in this case, the difference between RR and two other VM load balancers is so much and has the maximum response time obviously.

Other VM load balancing strategies since, even with more requests coming in, system performance won't suffer and the requests assigned to this datacenter will be fulfilled by the available VMs.

According to the simulation results, the RR algorithm performs better in this scenario than the previous one because, under this datacenter broker policy, the closet datacenter will receive the initial traffic. However, if response time starts to decline, the broker policy will split the load between the closet and the fastest datacenter. As a result, the round robin approach will perform better in this scenario by limiting the number of overloaded virtual machines.

D. ANALYTICAL BIRD'S-EYE VIEW

Throtteled load balancers provide the best VM load balancing performance in terms of the average total response time for all ClosetsP, OptP, and ReconfigP datacenter broker policies, as demonstrated by the simulation results shown in the preceding sections. In order to determine which is the best option, we thus analyze the performance of the three distinct broker policy combinations with the throttled load balancer. Figure 6 displays the outcomes of the experiment.



	100 bytes	500 bytes	1000 bytes	2000 bytes	5000 bytes
ClosestP-Thr	149.64	150.02	150.5	151.47	154.7
OptP-Thr	149.09	149.48	149.95	150.92	154.12
ReconfigP-Thr	149.88	184.91	153.66	160.12	505.62

Fig. 6. Comparison of Throttled VM load balancer with three different broker policies

The average response times of ClosestP-Thr and OptP-Thr are similar, as shown in Fig. 6, because both approaches use the same VM load balancing algorithm. The only difference is in the datacenter broker policies, where ClosestP and OptP behave similarly for initial traffic routing. However, we can conclude that the optimum approach is a combination of the closest datacenter broker policy and the throttled virtual machine load balancing algorithm based on the simulation result for longer task lengths. As we previously discussed, in ClosestP, the nearest datacenter is selected based on network latency. Requests are then sent to the closest resource, and the best outcome is achieved when these requests are handled by a throttled algorithm that keeps the VMs' performance from degrading.

Thus far, we have determined the optimal combination using the average response time parameter. The maximum and lowest response times for each of the nine combinations are displayed in Figure 7.

Tasks will receive resources at the outset of system requests without having to wait a long period, as shown in Fig. 7, where the minimum response time for all combinations is the same. Consequently, the same workload will be handled by the same resources in the same sequence. However, as we discussed in the previous part, ClosestP-Throtteled has the best average reaction time, hence it has the lowest maximum response time and the best maximum response time.

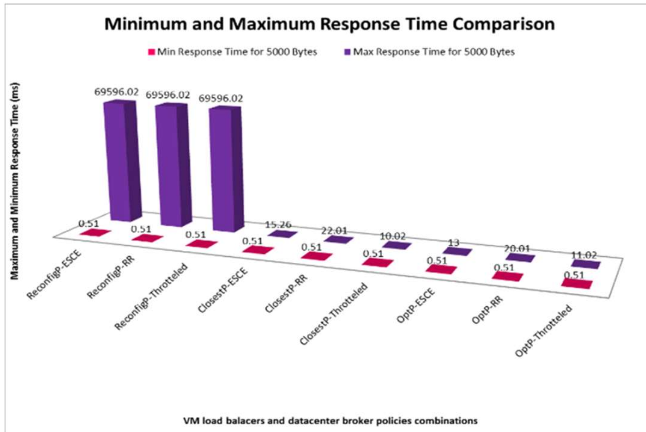


Fig. 7. Comparison of maximum and minimum response time for all 9 combinations

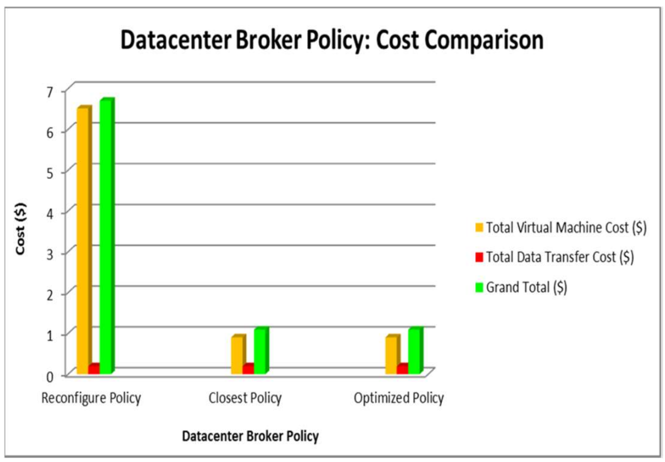


Fig. 8. Comparison of three datacenter broker polices cost

The cost-based performance evaluation of three datacenter broker strategies is displayed in Figure 8. The sum of the virtual machine and data transfer costs is known as the grand total. When compared to the cost of reconfiguring a policy, the closest and most optimal policies are the least expensive. According to our experimental results, the cost of data transfer is the same, but the total cost of virtual machines is higher in the Reconfigure policy because this policy tries to share the load of a datacenter and task with other datacenters; as a result, a task will be carried out by different VMs, resources, and consequently different and a higher price.

Based on the simulation results, we assessed the performance of various alternative configurations of datacenterbroker rules and VM load balancing algorithms, taking the outcome into account with various factors. Table II presents a comprehensive overview of the optimal combinations of virtual machine load balancing techniques, taking into account various parameter values.

TABLE II. BEST COMBINATIONS OF VM LOAD BALANICNG ALGORITHMS AND DATACENTER BROKER POLICIES

S.NO	VM Load Balancing Algorithm	Performance Evaluation Factors for Selecting Datacenter Broker Policy					
		Average Response Time (ms)		*Maximum Response Time (ms)*		*Total Virtual Machine Cost ($)*	
		Best Policy	*Simulation Result*	*Best Policy*	*Simulation Result*	*Best Policy*	*Simulation Result*
1	Round Robin	Optimize Response Time Policy	156.03	Optimize Response Time Policy	20.03	OptP / ClosestP	0.8
2	Throttled	Closest Data Center Policy	152.65	Closest Data Center Policy	10.04	OptP / ClosestP	0.8
3	Equally Spread Current Execution	Closest Data Center Policy	154.20	Closest Data Center Policy	14	OptP / ClosestP	0.8

CONCLUSION

In this paper, we examined the combinations of three distinct datacenter broker policies with three different load balancing methods in cloud computing environments: Round Robin, Throttled, and Equally Spread Current Execution VM. We suggested a simulated scenario to assess these load balancing strategies' effectiveness. We are able to build nine distinct load balancing ways by combining these combinations, and we simulated each approach for approximately five iterations with varying workloads. Finally, we obtain 45 distinct simulated outcomes from which we assess the cloud computing load balancing performance in terms of virtual machine cost, maximum and minimum response times, and average response times.

We used the CloudAnalyst simulator to simulate and analyze the performance of these methods. According to the simulation results, the throttled approach outperforms conventional load balancing techniques because it prevents overloaded virtual machines (VMs) from serving the workload by using a threshold and available VM list. Furthermore, we examined and recommended the optimal pairings between every virtual machine load balancer and the datacenter broker policy. In order to have a thorough survey, we will expand on these experimental results in the future by assessing more virtual machine load balancers in cloud computing and under other scenarios while taking into account more assessment criteria and characteristics.

REFERENCES

1. Jadeja, Y. and K. Modi. Cloud computing-concepts, architecture and challenges. in Computing, Electronics and Electrical Technologies (ICCEET), 2012International Conference on. 2012. IEEE.

2. Behl, A. Emerging security challenges in cloud computing: An insight to cloud security challenges and their mitigation. in Information and Communication Technologies (WICT), 2011 World Congress on. 2011.IEEE.

3. Mell, P. and T. Grance, The NIST definition of cloud computing. National Institute of Standards and Technology, 2009. **53**(6): p. 50.

4. Hong-hui, C., Cloud Computing Security Challenges. Computer Knowledge and Technology, 2011**. 24:** p. 014.

5. Li, J., et al., L-EncDB: A lightweight framework for privacy-preserving data queries in cloud computing. Knowledge-Based Systems, 2014.

6. PAULIESTHER, C.M., et al., TOWARDS SECURE CLOUD COMPUTING USING DIGITAL SIGNATURE. Journal of Theoretical and Applied Information Technology, 2015. **79**(2).

7. Oussalah, M., et al., Job scheduling in the Expert Cloud based on genetic algorithms. Kybernetes, 2014. **43**(8): p. 1262-1275.

8. Pop, F., et al., Deadline scheduling for a periodic tasks in inter-Cloud environments: a new approach to resource management. The Journal of Supercomputing, 2014: p. 1-12.

9. Dashti, S.E. and A. Masoud Rahmani, A New Scheduling Method for Workflows on Cloud Computing. International Journal of Advanced Research in Computer Science,2015. **6**(6).

10. Calheiros, R.N. and R. Buyya. Energy-efficient scheduling of urgent bag-of-tasks applications in clouds through DVFS. in 6th International Conference on Cloud Computing Technology and Science (CloudCom),. 2014.IEEE.

11. Gong, L., et al. Study on energy saving strategy and evaluation method of green cloud computing system. In Industrial Electronics and Applications (ICIEA), 2013 8th IEEE Conference on. 2013. IEEE.

12. Tadapaneni, N. R. (2020). Cloud Computing – An Emerging Technology. International Journal of Innovative Science and Research Technology. 5.

13. Dashti, S.E. and A.M. Rahmani, Dynamic VMs placement for energy efficiency by PSO in cloud computing. Journal of Experimental & Theoretical Artificial Intelligence, 2015: p. 1-16.

14. Xu, F., et al., Managing performance overhead of virtual machines in cloud computing: a survey, state of the art, and future directions. Proceedings of the IEEE, 2014. **102**(1): p. 11-31.

15. Anzt, H., Cojean, T., Yen-Chen, C., Dongarra, J., Flegar, G., Nayak, P., ... & Wang, W. (2020). Load-balancing sparse matrix vector product kernels on gpus. ACM Transactions on Parallel Computing (TOPC), 7(1), 1-26.

16. Van Chien, T., Björnson, E., & Larsson, E. G. (2020,May). Optimal design of energy-efficient cell-free massive MIMO: Joint power allocation and load balancing. In ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 5145-5149). IEEE.
17. Nuaimi, K.A., et al. A survey of load balancing in cloud computing: Challenges and algorithms. in Network Cloud Computing and Applications (NCCA), 2012 Second Symposium on. 2012. IEEE.
18. Kargar, M.J. and M. Vakili, Load balancing in MapReduce on homogeneous and heterogeneous clusters: an in-depth review. International Journal of Communication Networks and Distributed Systems, 2015. **15**(2-3): p. 149-168.
19. Tadapaneni, N. R. (2020). A Survey Of Various Load Balancing Algorithms In Cloud Computing. International Journal for Science and Advance Research in Technology, 6.
20. Alakeel, A.M., A guide to dynamic load balancing in distributed computer systems. International Journal of Computer Science and Information Security, 2010. **10**(6): p. 153-160.