Load Balancing Algorithms in Cloud Computing Analysis and Performance Evaluation

Mahnaz Hashemi
Department of Computer Engineering,
Science and Research Branch, Islamic
Azad University

Amir Masoud
Department of Computer Engineering,
Science and Research Branch, Islamic
Azad University

Abstract— **Distributing the system workload and balancing all incoming requests among all processing nodes in cloud computing environments is one of the important challenges in today cloud computing world. Many load balancing algorithms and approaches have been proposed for distributed and cloud computing systems. In addition the broker policy for distributing the workload among different datacenters in a cloud environment is one of the important factors for improving the system performance. In this paper we present an analytical comparison for the combinations of VM load balancing algorithms and different broker policies. We evaluate these approaches by simulating on CloudAnalyst simulator and the final results are presented based on different parameters. The results of this research specify the best possible combinations.**

***Index Terms*— Cloud Computing, *Virtual Machines, Load Balancing, Broker Policy, Performance Evaluation*.**

I. Introduction

Cloud computing platforms are growing in popularity rapidly these days. Cloud computing, often referred to as simply “the cloud”, is the delivery of on-demand computing resources over the Internet on a pay-for-use basis According to the official NIST definition, "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" [1]. Generally speaking, Cloud computing is a term for anything that involves delivering hosted services over the Internet. These services are mainly divided into three categories: Infrastructure-as-a-Service (IaaS), Platform-as-a-Service (PaaS) and Software-as-a-Service (SaaS) [2]. Most IT departments are forced to spend a significant portion of their time on frustrating implementation, maintenance, and upgrade projects. But now days, IT teams are turning to cloud computing technology for minimizing the time spent on lower-value activities and allow IT to focus on strategic activities with greater impact on the business. A cloud computing service has three main distinct characteristics that differentiate it from traditional hosting clearly. It is sold on demand, usually by the minute or the hour; it provides elasticity property which means that a client can have as much or as little of a service as they need at any given time and finally the services are fully managed by the cloud service providers. Apart from all of the cloud computing advantages, there are many challenges and open issues in cloud computing research areas such as: Security challenges [3-6], Job scheduling [7-10], Energy Efficiency and Green Computing [11-14] and Load Balancing [15-18].

Load balancing is one of the vital terms in cloud computing environments and generally distributed systems which affect the system performance dependent on the amount of work allocated to the system for a specific time period. Load balancing is the process of redistributing the general system workload among system resources for improving resource utilization and system performance [19]. Load balancing has been taken into consideration so that every virtual machine in the cloud computing system does the same amount of workload and therefore by increasing the throughput and minimizing the response time, users’ satisfactions will be provided.

In our approach, we present a performance evaluation and an analytical comparison between all common load balancing algorithms which are proposed and simulated in cloud computing simulator CloudAnalyst [20]. It enables users to evaluate requirements of large-scale Cloud applications in terms of geographic distribution in a quick and easy way [21]. We evaluate all the possible combinations of datacenter broker policy for distributing incoming jobs among available datacenters and load balancing mechanisms in each datacenter under the same comprehensive scenario. We will offer the best combination of these policies and load balancing mechanisms for having an analytical comparison by simulating all these different conditions.

The remaining sections of this paper discuss the following: Section II reviews some related works. In section

III we will explain the proposed scenario and some basic concepts about main datacenter broker policies and load balancing algorithms which are proposed on CloudAnalyst. Section IV shows the simulation results and makes an analysis of different combination of load balancing mechanisms and datacenter broker policies based on the simulation results. Finally in section V we will propose the conclusion and future work of this paper.

II. RELATED WORKS

There have been some works in load balancing performance evaluation and comprising different load balancing algorithms in cloud computing environments of which we will consider some in this section. In [22] a comparative study of two Round Robin and Throttled virtual machine load balancing algorithms has been proposed. In this study Round Robin and Throttled virtual machine load balancing policies are used along with optimized response time service broker policy and simulation is performed by adjusting parameters to inspect overall response time, datacenter hourly average processing times, datacenter request servicing time, response time according to region, user base hourly response times and total cost which has significant effect on performance. According to the simulation results, the combination of the proposed strategy of throttled and optimized response time service broker policy has the better performance than round robin load balancing algorithm in heterogeneous cloud computing environment.

Authors in [23] have presented a review of some load balancing algorithms in cloud computing for identifying qualitative components for simulation and analyzing the execution time of load balancing algorithms. In this study, the simulation process has been executed for three load balancing algorithms: Round Robin, Central queuing and Randomized with various combination of million instructions per second vs. VM an MIPS vs. Host. The simulation results show that response time is inversely proportionate with MIPS vs. VM and MIPS vs. Host, but optimum response time is achieved with same value of MIPS vs. VM and MIPS vs. Host.

A comparative study of three distributed load balancing algorithms for cloud computing scenarios has been proposed in [24]. In this study three representative algorithms were chosen for comparing performance evaluation. The first was directly based on naturally occurring phenomenon, honey bee foraging, the second sought to engineer a desired global outcome from biased random sampling, while the third used system rewiring which is called Active Clustering. The simulation results indicate that the honeybee-based load balancing algorithms give better performance when a diverse population of service types is required. In addition the simulation shows that random sampling walk performs better in confirming, similar populations and degrades quickly when the population diversity increases. Active Clustering perform better as the number of processing nodes is increased similar to random walk.

Authors in [25] discussed a performance comparison for different load balancing algorithms of virtual machine and policies in cloud computing. In this study four well known load balancing algorithms have been considered. Performance of Round Robin, Throttled, Execution Load and First Come First Serve Load Balancing Algorithms have been analyzed based on the average response time, average datacenter request servicing time and total cost. The simulation results according to the CloudAnalyst simulator show that round robin has the best integration performance.

III. PROPOSED SCENARIO, LOAD BALANCING ALGORITHMS AND POLICIES

In the previous section, we reviewed some related load balancing performance evaluation studies in cloud computing which have proposed simulation of VM load balancers. But all of the previous works just focused on load balancing in cloud datacenters while the way of distributing the workload among cloud datacenters which usually will be carried by datacenter brokers is so effective for balancing the loads and simulation results. In our approach we will consider the load balancing process in cloud computing in two different levels. In the first level which is presented by CloudAppServiceBroker in CloudAnalyst simulator, a model of service brokers has been proposed which handles traffic routing between user bases and datacenters. The three default and common routing policies which are provided in CloudAnalyst simulator are: “Closest Datacenter”, “Optimize Response Time” and “Reconfigure Dynamic with Load”. The second level which is introduced in CloudAnalyst by VMLoadBalancer component is responsible for modeling the load balance policy used by datacenters when serving allocation requests. There are three usual “Round Robin”, “Throttled” and “Equally Speared Current Execution Load” load balancing algorithms in each datacenter provided by simulator. By different combination of these three VM load balancing algorithms and datacenter broker Policies, nine different results are available which will be analyzed in the rest of this paper based on different evaluation parameters such as overall response time, datacenter processing time and cost. The remaining parts of this section will explain the simulated scenario, VM load balancing algorithms and datacenter broker policies.

A. Simulated Scenario

Figure 1 illustrates the simulated scenario in CloudAnalyst simulator. We use the same scenario for all different combinations of load balancing approaches to simulate under the same condition. As Fig. 1 shows the simulated scenario consists of two datacenters and three users which are placed in different geographical regions in the map. In region 0, there is datacenter 1 and there is no user base. R1 has just one user and no datacenter in this region while in region 5 there are one user and no datacenter and finally R4 which has one datacenter and one user base. By this kind of scenario configuration we tried to cover all possible situations for simulation process.

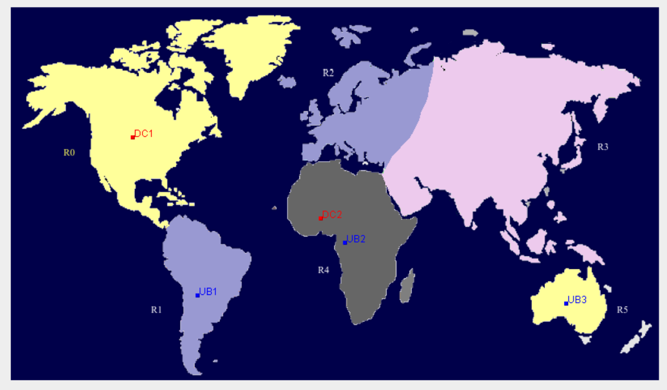

Fig. 1. The CloudAnalyst scenario on map

B. Datacenter Brocker Policies

Service broker policies handle traffic routing between user bases and datacenters. Three different datacenter broker policies have been implemented on CloudAnalyst simulator. The default routing policy which is called “Closest Data Center” policy (ClosestP) routes traffic to the closest datacenter in terms of network latency from the source user base. The second policy which is called “Optimize Response Time” policy (OptP), routes the Initial traffic to the closest to the requests originating in terms of network latency. Then if the response time achieved by the closest datacenter starts deteriorating, this service broker searches for the service broker with the best response time at the time and shares the load between the closest and the fastest data centers. The third load sharing mechanism which is called “Reconfigure Dynamically with Load” policy (ReconfigP) on CloudAnalyst attempts to share the load of a datacenter with other datacenters when the original datacenter’s performance degrades above a predefined threshold [20].

C. VM Load Balancing Algorithms

VM load balancing algorithms are used by datacenters when serving allocation requests for balancing the general workload in a datacenter. Several VM load balancing algorithms have been proposed in literature which three “Round Robin”, “Throttled” and “Equally Spread Current Execution Load” are implemented on CloudAnalyst simulator. In this section we introduce and explain briefly the general properties of these load balancing algorithms.

- Round Robin (RR)

  One of the simplest and well known scheduling and load balancing algorithms which utilize the principle of time slices is round robin algorithm [25]. Default load balancing algorithms on CloudAnalyst is round robin that allocates all incoming requests to the available virtual machines in round robin fashion without considering the current load on each virtual machine. This policy is not considered as priority intended scheduling policy. Large response time is a drawback in round robin architecture as it leads to degradation of system performance [26].

- Throttled

  Throttled algorithm initiates by assigning suitable virtual machine when clients send request to load balancer. This VM load balancing algorithm limits the number of requests being processed in each virtual machine to a throttling threshold [20]. The main role of throttled load balancer is to look after an index table of all virtual machine together with their states depicting busy and available mode. If client requests causing this threshold to be exceeded in all available virtual machines, the load balancer returns -1 value and datacenter queues the request until a virtual machine becomes available [22].

- Equally Spread Current Execution (ESCE)

  Equally spread current execution algorithm balances the tasks among available VM's in a way to even out the number of active tasks at any given time on each VM. ESCE algorithm handles the system workload with priorities [27]. ESCE distributes the datacenter workload randomly by checking the size and transfer the load to that virtual machine which is lightly loaded. This algorithm finds the VM with least number of allocations and in a way that the number of active tasks on each VM is kept evenly distributed among the VMs.

In the next section we will represent the simulation results of combination of these VM load balancing algorithms and datacenter broker policies. The main difference of our approach with literature review studies is simulating under a comprehensive and unique scenario and proposing a deep analytical comparison of several parameters of results.

IV. EXPERIMENTAL RESULTS AND ANALYTICAL COMPARISON

As we mentioned earlier we simulated the combination of different VM load balancers and datacenter broker polices under the same scenario which consist of two datacenters and three user bases in four different geographical points. Each datacenter include three physical servers and distribute the resources among its virtual machine based on time-shared policy. We execute the simulation duration about 60 minutes for each iterate.

TABLE I. SIMULATION CONFIGURATION SUMMARY

#	Cloud Resources	Number of processor per each Physical Server	Cost per VM ($/Hr.)	OS / Arch	VMM	Data Transfer Cost ($/Gb)
1	Datacenter 1 (Region 0)	4	0.1	Linux / X86	Xen	0.1
2	Datacenter 2 (Region 4)	4	0.1	Linux / X86	Xen	0.1

We simulate 9 different load balancing approaches under the same scenario. We increase the cloudlet lengths from 100 to 5000 bytes in 5 steps and therefore simulated 45 different simulation iterates. The table I shows the simulation process in detailed. We will analyze the simulation results at the remaining parts of this section. Figure 2 shows the delay latency matrix which is used by datacenter broker policies for selecting the target 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. Case1: Closet Data Center Policy (ClosestP)

In case 1, we select ClosestP as the datacenter broker policy and simulated the same workload with three RR, Throttled and ESCE VM load balancing algorithms. Figure 3 illustrates the average response time of total datacenters and user bases.

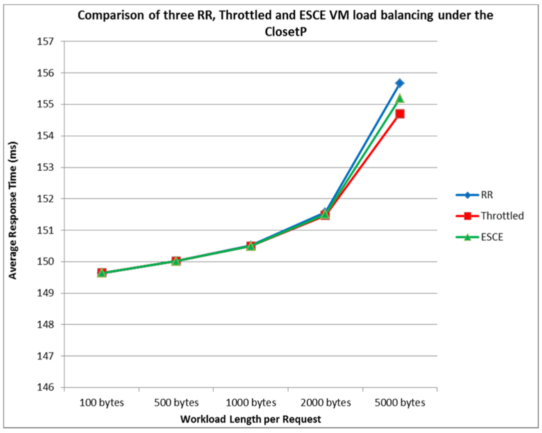

Fig. 3. Comparison of three RR, Throttled and ESCE VM load balancing under the ClosetP

As it is shown in Fig. 3, in this case the Throttled load balancing algorithm has the best response time than the others in combination with closets datacenter policy as the volume per request of datacenter workload increases. As the workload increases the possibility of having the under loaded and overload virtual machines will be increased by distributing the workload randomly. So In this situation RR algorithm doesn't work so optimized as the result shows, because it distributes the load among system nodes without any consideration about their current loads. But the throttled algorithm keeps all virtual machines load in a normal state by using the throttled threshold any preventing sending the job requests to the VMs which have some jobs to process.

Therefore by using the Throttled algorithm the system performance won't degrade and in the situation of large amount of incoming requests will have a better average response time. The ESCE algorithm consider the number of allocated tasks to each virtual machine and based on that distribute the future work load among the VMs but it doesn't consider the workload length. Then it has a better performance than RR, but because it doesn't care about VMs' workload it doesn't work as well as Throttled load balancer.

B. Case2: Optimize Response Time Policy (OptP)

In case 2, the OptP has been chosen as the datacenter broker policy and simulation process has been executed with the same workload with three RR, Throttled and ESCE VM load balancing algorithms. Like previous policy this broker policy finds the destination datacenter based on the delay matrix at the first. Figure 4 shows the average response time of total datacenters and user bases.

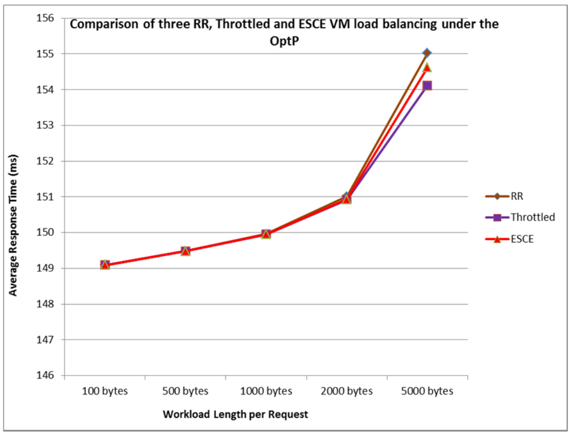

Fig. 4. Comparison of three RR, Throttled and ESCE VM load balancing under the OptP

As it is shown in Fig. 4, the simulation results here is similar to previous case. Again Throttled algorithm has the best performance in terms of total average response time than other VM load balancing algorithms because under the increasing incoming requests, the system performance won't degrade and available VMs will serve the request which allocated to this datacenter.

The simulation results show that RR algorithm has the better performance in this case than previous one because in this datacenter broker policy the initial traffic will be routed to the closet datacenter, but if response time starts deteriorating, this broker policy shares the load between the closet and the fastest datacenter. Therefor in this case the round robin algorithm will have the better performance by preventing the occurrence of more overloaded VMs.

C. *Case3: Reconfigure Dynamically with Load Policy (ReconfigP)*

In case 3, we simulated the combination of ReconfigP with three VM load balancing algorithms for 15 iterations like previous situations. Figure 5 Shows the simulation results which had some unexpected variations.

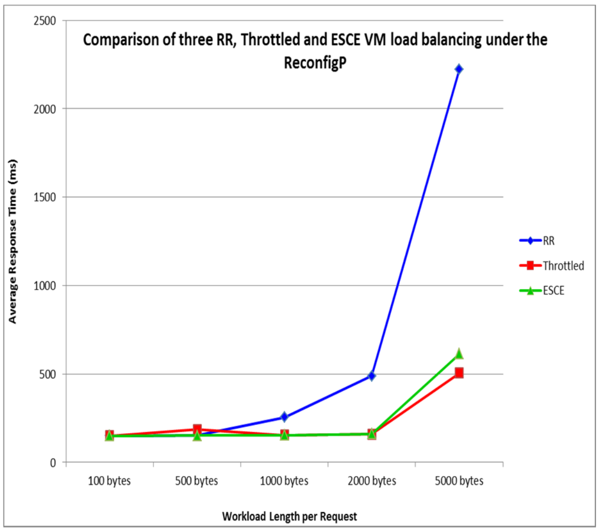

Fig. 5. Comparison of three RR, Throttled and ESCE VM load balancing under the ReconfigP

The results for this case are very different in comparison with two previous cases. While the Throttled algorithm still offers a better performance, but there is an unexpected variation between workload with length 500bytes and 1000 bytes. The dynamic reconfiguration policy couldn't offer a suitable configuration and share the load of one of the datacenters with other one; therefore in the case of Throttled algorithm and when the workload length is closing to 500 byes a large amount of workload was imposed to the datacenter which caused a degradation performance. In addition in this case the difference of RR and two others VM load balancers are so much and have the maximum response time obviously that we can say the combination of RR and ReconfigP has the worst result for this VM load balancing algorithms because the increasing amount of system workload and balancing it by RR without considering the current VMs' load caused to heavy overloaded situation.

Other VM load balancing algorithms because under the increasing coming request, the system performance won't degrade and available VMs will serve the request which allocated to this datacenter.

The simulation results show that RR algorithm has the better performance in this case than previous one because in this datacenter broker policy the initial traffic will be routed to the closet datacenter, but if response time starts deteriorating, this broker policy shares the load between the closet and the fastest datacenter. Therefore in this case the round robin algorithm will have the better performance by preventing the occurrence of more overloaded VMs.

D. *ANALYTICAL BIRD'S-EYE VIEW*

As the simulation results illustrated in previous sections, the best VM load balancing pefromance in terms of average of total response time for all ClosetsP, OptP and ReconfigP datacenter broker policy, belongs to Throtteled load balancers. Therefore we compare the performance of the three combinations of different broker policies and Throttled load balancer for finding the best solution. Figure 6 shows the experimental results.

	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

As Fig. 6 shows, the ClosestP-Thr and OptP-Thr have the similar and approximatley same average response time, because in both approach the VM load balanicng algorithm is same and the diffren is just in datacenter broker policies that ClosestP and OptP have the same behaviour for the initial traffic routing. However based on the simulation result for larger workload length we can say that the best solution is using combination of closest datacenter broker policy and Throttled VM load balancing algorithm. In ClosestP as we mentioned earlier the closest datacenter will be choosen based on the network latency and just sending request to the closest resource and when handling these request by throttled algorithm which prevents the VMs' performance degradeation the best result will be achived.

Till now we evalute the best combination based on the average response time parameter. Figure 7 shows the maximum and minimum response time for all nine combinations.

As it is illustrated in Fig. 7, the minimum response time for all combinations is same and it is because of this fact that at the first requests in the system, tasks will get resources without any considerable waiting time. Therefore same workload will be served in the same order and by same resources. But the best maximum response time belongs to ClosestP-Throtteled which has the least maximum response time because this approach has the best average response time as we explained in previous section.

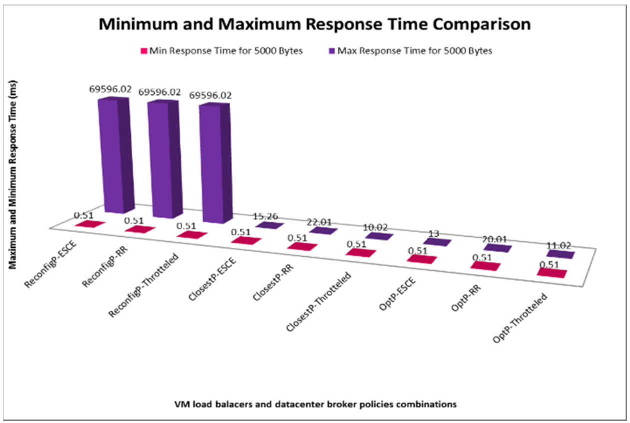

Fig. 7. Comparison of maximum and minimum response time for all 9 combinations

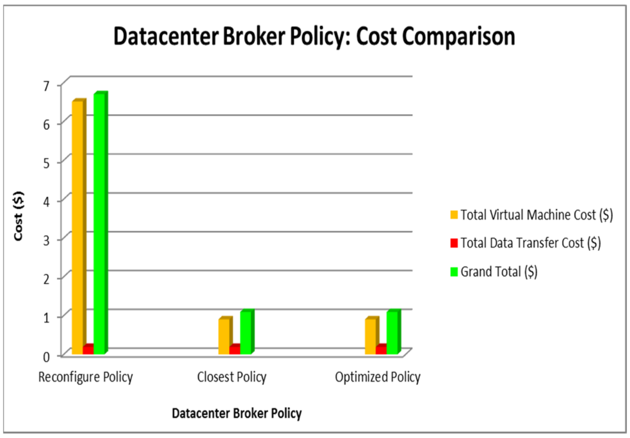

Fig. 8. Comparison of three datacenter broker polices cost

Figure 8 shows the performance evaluation of three datacenter broker policies in terms of cost. The Grand total is the total of virtual machine cost and data transfer cost. The Closest policy and optimized policy have the least costs in comparision with Reconfigure policy cost. The cost of data transfer based on our experimental results are same but the total virtual machine cost is more expensive in Reconfigure policy because this policy try to share the load of a datacenter and task with other deatcenters and therefore a task will be executed by different VMs, resources and therefore different and more expensive cost.

We evaluated the performance of different possible combinations of VM load balancing algorithms and datacenterbroker policies based on the simulation results and considered the result through different parameters. In table II we proposed a general review of the best combinations of the VM load balancing algorithms in terms of different parameters.

V. CONCLUSION

In this paper we analysed the combiations of three Round Robin, Throttled and Equally Spread Current Execution VM load balancing algorithms and three different datcenter broker policies in cloud computing environments. We proposed a simulation scenario for evaluating the performance of these load balning approaches. By these combinations, we generate the nine different possible load balancing approaches which simulated each one about five iterations with dirfferent workloads. Finally we achive 45 different simulated results that throgh these results we compare the performace of load balancing in cloud compuing in terms of average response time, maximum and minimum response time and virual machine cost.

We Analysed the performance of these approaches by simulating on CloudAnalyst simulator. The simulation results shows that throttled algorithm have a better peformance than other load balancing algorithms, because it usese a threshold and available VM list for preventing serve the workload by overloaded VMs. In addition we analysed and offered the best combinations of each VM load balancer with datacnter broker policy. As the future works we will exapnd these experimental results by evaluating the more VM load balancers in cloud computing and under the different scenarios by considering the more evaluation factors and parameters for having an comprehensive survey.

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

#	VM Load Balancing Algorithm	Performance Evalauation Factors for Selecting Datacnter 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	155.02	Optimize Response Time Policy	20.01	OptP / ClosestP	0.9
2	Throttled	Closest Data Center Policy	154.70	Closest Data Center Policy	10.02	OptP / ClosestP	0.9
3	Equally Spread Current Execution	Closest Data Center Policy	155.19	Optimize Response Time Policy	13	OptP / ClosestP	0.9

REFERENCES

1. Mell, P. and T. Grance, *The NIST definition of cloud computing.* National Institute of Standards and Technology, 2009. **53**(6): p. 50.
2. Jadeja, Y. and K. Modi. *Cloud computing-concepts, architecture and challenges*. in *Computing, Electronics and Electrical Technologies (ICCEET), 2012 International Conference on*. 2012. IEEE.
3. 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.
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 aperiodic 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.
21. Wickremasinghe, B., R.N. Calheiros, and R. Buyya. *Cloudanalyst: A cloudsim-based visual modeller for analysing cloud computing environments and applications*. in *Advanced Information Networking and Applications (AINA), 2010 24th IEEE International Conference on*. 2010. IEEE.
22. Goga, K., et al. *Simulation, modeling and performance evaluation tools for cloud applications,"*. in *Proceedings of the 8th International Conference on Complex, Intelligent and Software Intensive Systems (CISIS2014)*. 2014.
23. Behal, V. and A. Kumar. *Cloud computing: Performance analysis of load balancing algorithms in cloud heterogeneous environment*. in *Confluence The Next Generation Information Technology Summit (Confluence), 2014 5th International Conference-*. 2014. IEEE.
24. Ray, S. and A. De Sarkar, *Execution analysis of load balancing algorithms in cloud computing environment.* International Journal on Cloud Computing: Services and Architecture (IJCCSA), 2012. **2**(5).
25. Randles, M., D. Lamb, and A. Taleb-Bendiab. *A comparative study into distributed load balancing algorithms for cloud computing*. in *Advanced Information Networking and Applications Workshops (WAINA), 2010 IEEE 24th International Conference on*. 2010. IEEE.
26. Mohapatra, S., K. Smruti Rekha, and S. Mohanty, *A Comparison of Four Popular Heuristics for Load Balancing of Virtual Machines in Cloud Computing.* International Journal of Computer Applications, 2013. **68**(6): p. 33-38.
27. Singh, A., P. Goyal, and S. Batra, *An Optimized Round Robin Scheduling Algorithm for CPU Scheduling.* IJCSE) International Journal on Computer Science and Engineering, 2010. **2**(07): p. 2383-2385.
28. Nitika, M., M. Shaveta, and M.G. Raj, *Comparative analysis of load balancing algorithms in cloud computing.* International Journal of Advanced Research in Computer Engineering & Technology, 2012. **1**(3): p. 120-124.
29. Mesbahi, M., A.M. Rahmani, and A.T. Chronopoulos. *Cloud light weight: A new solution for load balancing in cloud computing*. in *Data Science & Engineering (ICDSE), 2014 International Conference on*. 2014. IEEE.
30. Hsu, C.-H., et al. *Energy-aware task consolidation technique for cloud computing*. in *Cloud Computing Technology and Science (CloudCom), 2011 IEEE Third International Conference on*. 2011. IEEE.
31. Jain, A., et al. *Energy efficient computing-Green cloud computing*. in *Energy Efficient Technologies for Sustainability (ICEETS), 2013 International Conference on*. 2013. IEEE.