Algorithm Simulation of an Efficient Strategy for Green Cloud Computing

Moses Kazeem Abiodun (moses.abiodun@lmu.edu.ng)
Landmark University https://orcid.org/0000-0002-3049-1184

Research Article

Keywords: Cloud Computing, Green Cloud, Environment, Load Balancing, Service Broker, Energy, SDG 7 and SDG 4

Posted Date: May 9th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-910771/v1

ALGORITHM SIMULATION OF AN EFFICIENT STRATEGY FOR GREEN CLOUD COMPUTING

Abiodun K. Moses[1], Awotunde Joseph[2], Adeniyi Emmanuel[1], Adebiyi Ayodele A.[1], Aremu Rueben[2], Chieke Israel[1]

[1]Department of Computer Science, Landmark University, Omu-Aran.

[2]Department of Computer Science, University of Ilorin, Ilorin.

moses.abiodun@lmu.edu.ng, awotunde.jb@unilorin.edu.ng , adeniyi.emmanuel@lmu.edu.ng, ayo.adebiyi@lmu.edu.ng, draremu2006@gmail.com, chieke.israel@lmu.edu.ng

ABSTRACT

In order to reduce the energy costs of running cloud computing services, this study looked into the methodologies and algorithms used in Green Cloud Computing (GCC). Huge datacenters and network transferring services need a lot of energy, which results in a lot of CO2. As a result, the GCC concept has arisen and evolved in order to reduce electrical power energy consumption and worldwide carbon dioxide emissions while also maintaining the services' long-term sustainability, dependability, and efficiency. In this research work, Cloud Analyst simulation software was used to mimic six datacenters in regular operation for six users in diverse locations around the world during a 365-day period. The TLB and ORTP Service Broker algorithm combination is the ideal algorithm combination because it has the fastest reaction time, which means it uses fewer resources and equipment, saving power and money. On the other side, the ESCELB Policy and ORTP Broker is the worst algorithm combination since it takes the longest time to respond, consumes more power, and increases overall cost. As a result, the study concludes that TLB and ORTP strategies should be implemented for effective green cloud computing in a safer environment.

Keyword: Cloud Computing, Green Cloud, Environment, Load Balancing, Service Broker, Energy, SDG 7 and SDG 4

1.1 Background of Study

Cloud computing can be said to be a new paradigm for Dynamic data-supported computer service provisioning centers that commonly uses the technology of virtual machines (VM) for consolidation [1]. Cloud computing grants its users around the world easy access to resources with minimal maintenance, shared pools of services and resources such as servers, applications, monitoring, databases, and more from a centralized location.

There are three types of cloud computing models: Platform-as-a-Service (PaaS), Software-as-a-Service (SaaS), and Infrastructure-as-a-Service (IaaS). Software as a service (SaaS) is a software licensing and delivery model in which software is licensed on a subscription basis and is centrally hosted [2]. Platform-as-a-Service (PaaS), also known as application platform as a service (PaaS), or platform-based service, is a type of cloud computing service that provides a platform for customers to develop, run, and manage applications without constantly worrying about the infrastructure that comes with developing and launching an app. [3]. Infrastructure as a service (IaaS) are online services that provide high-level APIs used to dereference various lowlevel details of underlying network infrastructure like physical computing resources, location, data partitioning, scaling, security, backup etc.

Green technology has become a major concern for governments and corporations all across the world because of its environmental, economic, and political aspects. Due to the increase in internet usage and the various advantages that the cloud offers such as scalability, increased collaboration, fault tolerance, and high availability, companies are starting to shift to using Cloud services.

Load balancing is one of the most difficult aspects of cloud computing [4]. It is a technique in which distribution of the complex local workload is uniform across the nodes in cloud in order to avoid the situation where few nodes are overloaded while few are dormant.

When creating their own private cloud, businesses can choose from a variety of cloud computing implementations. Several potential alternatives are Open Nebula, Nimbus, and Cloudbus.

Green computing also called ICT sustainability is the study of developing and supplying computer resources like servers and accompanying subsystems like monitors and network gadgets in order to have a lower environmental impact. [5].

The world is moving to green cloud computing because it allows us to help people save energy, decrease their carbon footprint, and move toward a greener, smarter future, and we are utilizing the Load Balancing Technique to do it. Load balancing allows us to help reduce energy consumption.

Load balancing aims to improve cloud performance by speeding up different limited characteristics such as reaction time, execution time, system dependability, and so on. Load balancing algorithms are classified into two areas depending on the device state: static and dynamic algorithms.

When distributing the load, a static load balancing algorithm ignores the past state or activity of a node. Static load balancing methods, on the other hand, have the disadvantage that tasks are allocated to the processor or machines only after they are generated, and thus jobs cannot be transferred to another machine for load balancing during execution. [6].

A dynamic load balancing method, on the other hand, evaluates the prior condition of a node when distributing the load, such as CPU load, memory used, latency, and network load, among other factors [7]. The current status of the system is utilized to carry out load management decisions. Because the current state of the system is utilized to make dynamic load balancing choices, processes are allowed to move from an overloaded computer to an underused machine in real time [8]. The other part of this paper includes section two that review related works of literature; section three discusses the approach and the performance metrics adopted for the study. Section four shows how the simulation was carried out and the simulation outcomes and discussion. Lastly, Section five recapitulates the research outcomes and proposal for future work.

1.2 Green Cloud Computing

Green cloud computing refers to the creation, generation, and use of digital spaces in an ecologically responsible manner. A green cloud may save energy while also making better use of resources in cloud computing. Green cloud computing allows users to benefit from the benefits of cloud storage while reducing negative consequences on the environment, which has an influence on humanity's well-being.

Importance of green cloud computing?

Green cloud computing is a system that allows users to take advantage of all the advantages of cloud storage while reducing its negative impact. The ultimate objective of green cloud computing is to minimize big data center power usage. Some of the following are other importance of green cloud computing:

I. Reduced energy usage
II. Cost effective
III. Simpler maintenance
IV. Less taxing to the environment due to lower carbon emission.

1.3 What is load balancing in cloud computing?

In cloud computing, load balancing is a technique or approach for distributing dynamic local workload equitably among nodes in the cloud to avoid situations where a few nodes are overburdened while others are idle. As a result, the quantity of energy consumed is reduced [9]. The following are some of the various load balancing techniques prevalent in cloud computing:

a. Dynamic Round-Robin algorithm
b. Power Aware load balancing(PALB)
c. Equally Spread Current Execution algorithm(ESCE)
d. Minimum Cost Maximum Flow algorithm(MCMF)
e. Task Consolidation algorithms
f. Min-Min algorithm
g. A hybrid algorithm
h. Join-Idle-Queue

The existing load balancing approaches that have been researched mostly focus on lowering overhead, improving service response time, and increasing performance, among other things. As a result, a hybrid load balancing approach is required to increase cloud computing performance while maximizing resource usage and, as a result, lowering energy consumption.

2.0 Review of Related works

[10], a review of cloud computing load balancing strategies was presented. The author splits job scheduling and load balancing algorithms into seven categories: Hadoop map reduces load balancing, proxy-based load balancing, natural phenomenon-based load balancing, Application Oriented Load, General Load Balancing, Network Aware Load Balancing, and Workflow Specific Load Balancing. Load balancing is classified into two domains in the literature based on the condition of the system and the startup procedure. Different algorithms are classified into categories, with pros and downsides listed for each.

At the same time, [11] analyzed the load balancing technology already recognized based on the survey; the current algorithms were categorized into three categories by the author: static, dynamic, and hybrid. The author codified the difficulties surrounding load balancing and answered the most important issues, such as the desired level of indications, responsibilities, and load balancing obstacles. Follow the proper search operations in the search query, apply Boolean operations on the search string to help obtain the most relevant material from multiple publication sources, and utilize the Quality Assessment Checklist (QAC) to conduct the selection criteria stage. However, just a few metrics of QoS were investigated in these two surveys, including response time, lead time, scalability, resource usage, migration time, performance, and power savings. When considering other important QoS indicators (such as cost of migration, service level violation, grade). Balance, task rejection rate, etc.

[12] League Championship Algorithm (LCA) supervised a comparative analysis of different programming algorithms for cloud computing and network computing, taking into account five main meta-heuristic approaches, including ant colony optimization (ACO), genetic algorithm (GA), and particle swarm optimization (PSO). In addition, the BAT algorithm. Furthermore, an entire technology comparison is presented; however, his study is confined to programming algorithms victimization meta-heuristic approaches. What is more, the survey is proscribed to organic process algorithms and doesn't give a large categorization.

[13] studied the essential demands, design, and implementation features of load balancers for cloud providers, categorizing them into three categories: general algorithm-based, architecture-based, and artificial intelligence-based. As with earlier studies, this one uses static and dynamic categorization as broad categories. However, the author points out that building load balancing

algorithms is quite challenging. Based on his research into dynamic, distributed, and non-cooperative algorithms, the author made a judgment.

[14] A categorization article on current load balancing methods, such as those mentioned in the preceding study, recommended that these algorithms be separated into static and dynamic algorithms. They also learned about the difficulty of solving the load balancing problem. Challenges include geographic distribution of nodes, migration time, system performance, power management, and security. These issues have long been listed in the literature. In fact, the author compares existing load balancing algorithms based on certain QoS indicators (such as performance, speed, response time, migration time, etc.). The document concluded that there is a trade-off between indicators. The limitation of this paper is that only eight load balancing algorithms are compared among a large number of algorithms.

[15] Discussed a comprehensive overview of load balancing algorithms. Different load balancing methods are divided into static and dynamic according to the system state, and divided into homogeneous and heterogeneous according to the consistency of the VM type. Performance indicators are also used to rank load balancing methods. In addition, the advantages and disadvantages of each algorithm are discussed.

[16], has performed a well-defined, systematic, and prospective assessment of load balancing solutions in software-defined networks, and has generally categorized them into deterministic and non-deterministic approaches, with related indications under research. The study presents a number of significant issues and attempts to answer them based on their relevance, measurement analysis, functions, and problems in software-defined network load balancing. The author's study examines the benefits and drawbacks of current communication network literature. Also, although not being particular to the cloud computing area, the paper has a solid basis and a good link

between load balancing metrics. Furthermore, rather of using hierarchical classification, the study uses single-level categorization. Standards are set for classification purposes, but the characteristics of generalization and specialization are not drawn, which ultimately leads to inadequate and inadequate conclusions. Additionally, previous review papers did not examine several critical characteristics, such as the load balancing algorithm's complexity and the percentage of load balancing indications in the literature. Existing survey materials do not include a comprehensive explanation of the QoS indicator set, and additional indications are anticipated (such as migration costs, service level violations, balance introduced in the investigation, task rejection rate). In this paper, a taxonomy-based categorization is presented and its validity is tested against current literature. As a result, a classification system based on categorization was developed to help future researchers in developing an improved, efficient, robust, and fault-tolerant load balancing algorithm, as well as to offer them with fresh insights into future work. The suggested categorization technique is based on various aspects of the load balancing algorithm, including "the nature of the algorithm," "the state of the algorithm," "load balancing features," "execution mode," "type," "function," and "algorithm use Technology."

[17], present a distributed architecture and illustrate how the utility functions can enable the collection of decentralized elements to continuously optimize the use of computational resources in a complex environment. To achieve this, the authors used a functional prototype data center. Architecture is typically a two-tier system with separate, autonomous components that encourage versatility. The authors present statistical data showing the effectiveness of the scheme in handling realistic, fluctuating web-based transactional workloads operating on a Linux cluster.

[18], discussed how reorganizing virtualized servers into fewer physical servers leads to flexible resource provision and improved energy efficiency. A model for estimating each virtual machine's

energy consumption without dedicated metering hardware. The authors also proposed a virtual machine scheduling algorithm based on an estimation model. According to the energy budget of each virtual machine, the algorithm provides computing resources. Xen virtualization system was used for implementing the model, and an estimated error of less than 5 percent of the total energy consumption and provided energy consumption was recorded.

[19], propose an architecture called GENiC, an integrated energy management system that addresses this issue for data center wise optimization. It captures monitoring, cooling, waste heat recovery, control of the IT workload, and local power generation to achieve energy efficiency. The optimization happens at different estimated lengths of times: short term predictions and long term predictions. The authors also propose a strategy to optimize the allocation of Virtual Machines (VMs) while unused servers are switched off. Virtual Machine (VM) migration is a technique that has recently emerged in data centers. It shifts the virtual machine from one physical hardware to another in case of unscheduled server downtime or because of some fault in the server. This is done so that there is high availability. VM migration can also be performed for disaster recovery purposes.

[20], purpose an algorithm that efficiently allocated VM requests to the physical nodes in a best-fit strategy. This goal is attained by designing a performance analysis scheme for each node which takes into consideration the specifications on cores, CPU, and memory.

[21], proposes a new load balancing algorithm, which combines maximum-minimum and round-robin (MMRR) algorithm The study findings indicate that MMRR has brought significant changes to cloud services. MMRR performed better from the algorithms tested based on the whole response time and cost-effectiveness (89%). The study suggested that MMRR should be implemented for enhancing user satisfaction in the cloud service.

3.0 Methodology

This research looks at how to create an effective green cloud computing strategy utilizing a cloud analyst, a framework for modeling and simulation of cloud computing infrastructures and services, in order to minimize excessive energy usage in servers.

This was achieved using some selected algorithms which falls into two categories;

A. Load Balancer Algorithm
   i. Round Robin Load Balancing (RRLB)
   ii. Throttled Load Balancing (TLB)
   iii. Equally Spread Current Execution Load Balancing (ESCELB)

B. Service Broker Algorithm
   i. Optimize Response Time Policy(ORTP)
   ii. Reconfigure Dynamically with Load Policy(RDLP)

Performance Evaluation Parameter

There are various factors that need to be considered for energy efficiency in data centers. The first step to green data centers is to define certain green metrics based on which algorithms can be quantified. The most common metric is Power usage efficiency (PUE) which is calculated by total system energy to energy used by data center equipment.

The servers are the primary cause of energy use in data centers. Usually, only 30 percent of the time servers are busy. The resource usage of the computer server is equal to the use of the CPU. The idle server uses approximately two-thirds of its peak load usage to maintain memory, disks, and I / O services running, while one-third of its usage decreases linearly with the use of CPU. All servers available are either in mode or idle to respond to the immediate peak load.

These parameters can be used to evaluate the suggested approach's performance.

I. Overall Response Time: The user base request completion time and datacentre processing time have both been measured. These parameters show how long it takes for a cloud service provider to reply to a request. The earlier the better for saving energy usage and cost.

II. Datacentre Request Servicing Time: Datacentre request servicing time gives information on a single datacentre's ability to service several user bases' requests.

3.1 GCC Simulations of an E-commerce Website using Cloud Analyst

This study examines a business-to-customer e-commerce model (websites like Jumia, Amazon, Gonga, etc.) that is housed in six data centers throughout the globe, each with a several amount of virtual machines, like 50, 100, and 150 VMs respectively. Each VM's image is roughly 10000Mb in size, with 512kb of RAM and a bandwidth of 1000 Mbit/s. Every user is limited to 20 requests per hour throughout the course of a 365-day experiment.

The study simulates the outcomes using six distinct cases, then compares the results using six alternative algorithm combinations to find the optimal method for improving the GCC concept while reducing power consumption and CO_2 emissions.

3.2 Simulation Setup

Table1 shows the parameters used for simulation.

Table 1: Variables and values used in simulation

Variables	Value Used
Usable BW per Machine in the Data Center	2000
Storage per machine in a data center	100000
Bandwidth for Virtual Machines	2000
Virtual Machines memory	512Kb
VM Policy of Data Center	Time Shared

As a result, table 2 illustrates the algorithm simulation carried out; each case comprises a mixture of two algorithms that are utilized each time in simulation.

Table 2: Cases and algorithms employed in simulation

Case	Algorithm
1	The RRLB and ORTP Service Broker
2	The RRLB and RDLP Broker
3	The TLB and ORTP Service Broker
4	The TLB and RDLP Broker
5	The ESCELB and ORTP Service Broker
6	The ESCELB and RDLP Broker

The simulation results for six cases involving 50, 100, and 150 virtual machines, separately.

4.0 RESULTS AND DISCUSSION

The outcomes given the different algorithms of the different scenarios after simulation are display in Tables 3 - 5:

4.1 Results when using 50VMs

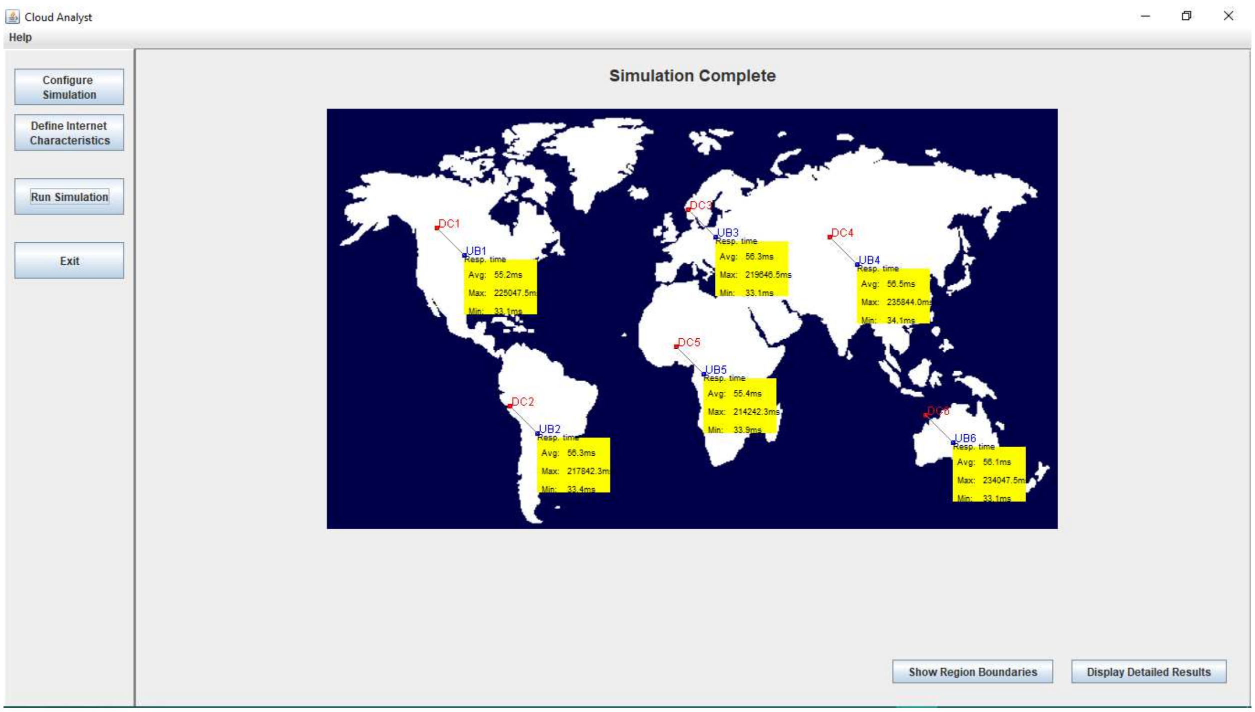

Figure 1: Case 1 for 50VMs simulation result

Table 3: Shows the result from several cases using 50 VMs

Case	Overall Response Time	Data Center Processing Time
1	55.88	6.24
2	55.90	6.25
3	50.11	0.48
4	50.13	0.49
5	55.98	6.35
6	55.95	6.33

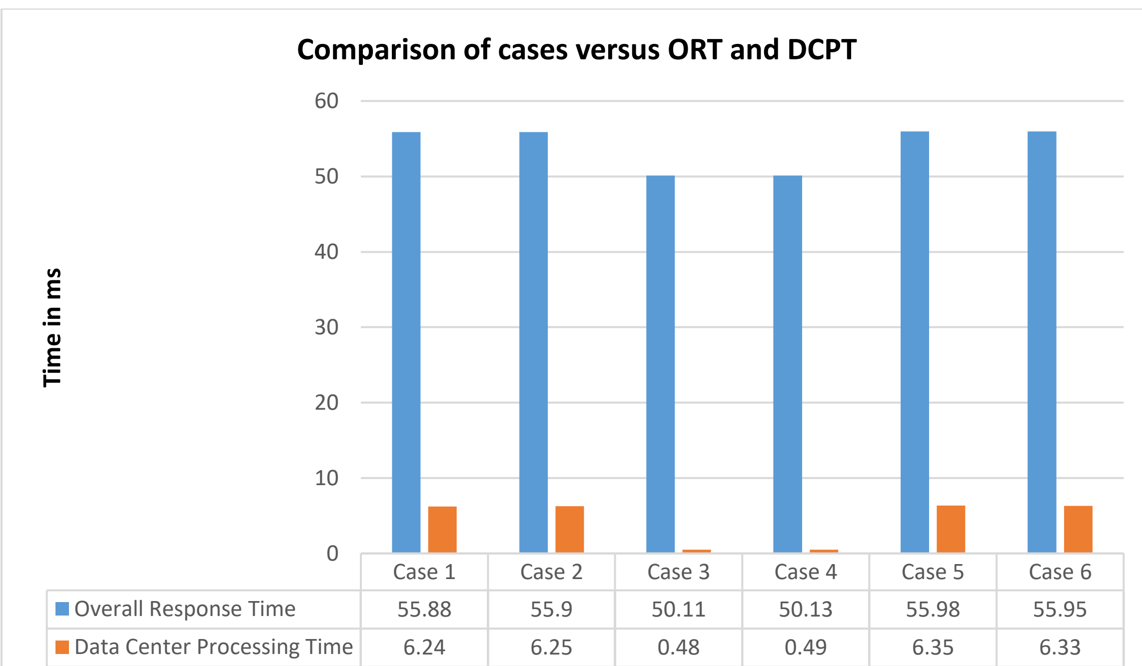

Figure 2: Comparison of cases versus ORT and DCPT

As seen in Table 3 and Figure 2, the combination of the Throttle Load Balancing techniques and the optimized response time service broker policy shows a better response time for both overall response time and data center processing time. The ESCELB and ORTP Service Broker on the other hand, are the poorest combination since it takes the longest time to respond.

4.2 Results when using 100VMs

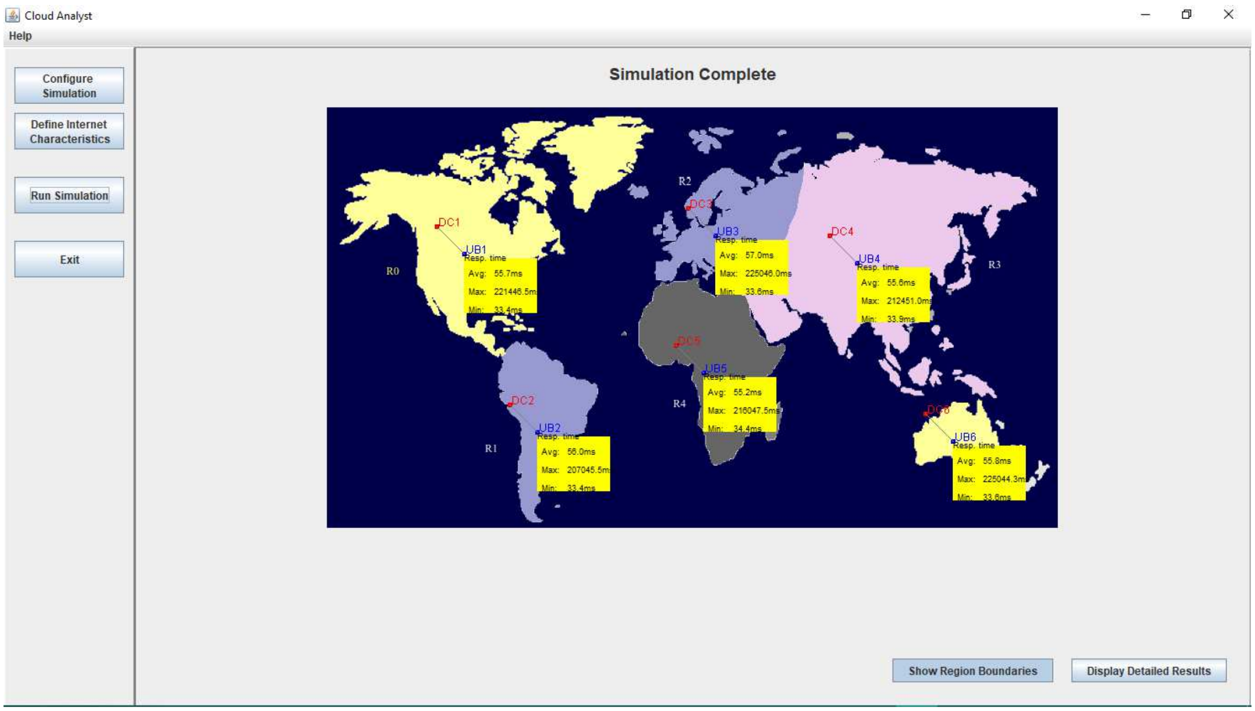

Figure 3: Case 1 for 100VMs simulation result

Table 4. Shows the result from several cases using 100 VMs

Case	Overall Response Time	Data Center Processing Time
1	55.92	6.29
2	55.95	6.32
3	50.14	0.54
4	50.17	0.55
5	56.02	6.37
6	55.89	6.26

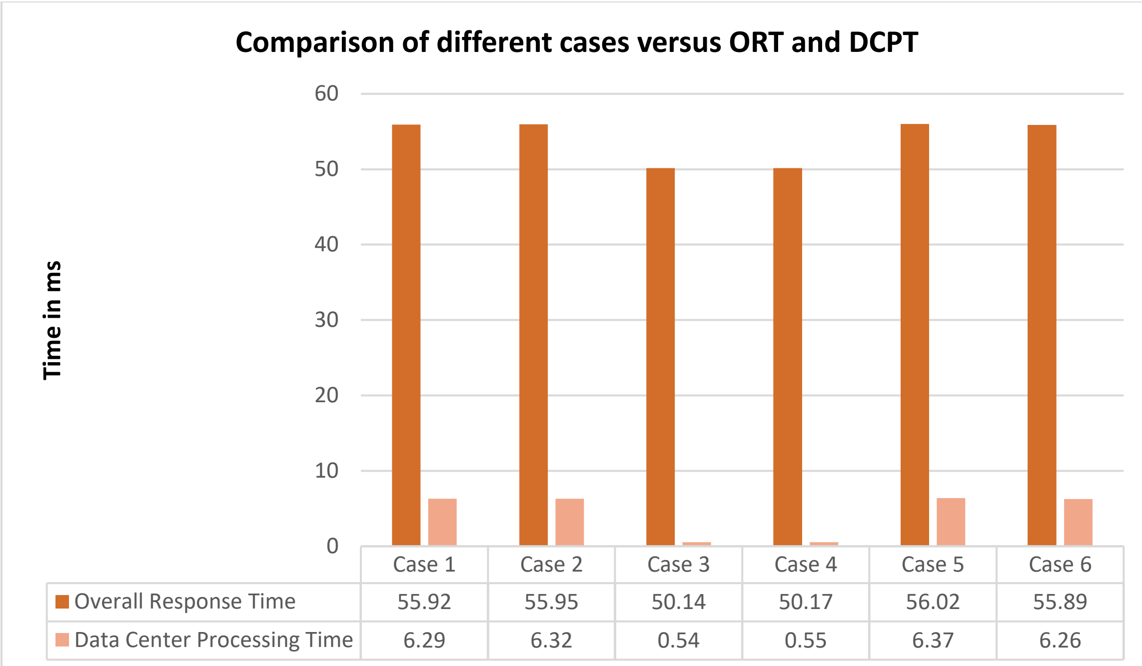

Figure 4: Comparison of cases versus ORT and DCPT

As seen in Table 4 and Figure 4, result shows that the finest algorithms are the combination of TLB Policy and ORTP service broker. This came as a result of the finest RT service broker choosing the DCs with the minimum RT. The worst combination, on the other hand, is ESCELB and ORTP Broker, which has the longest reaction time.

4.3 Results when using 150VMs

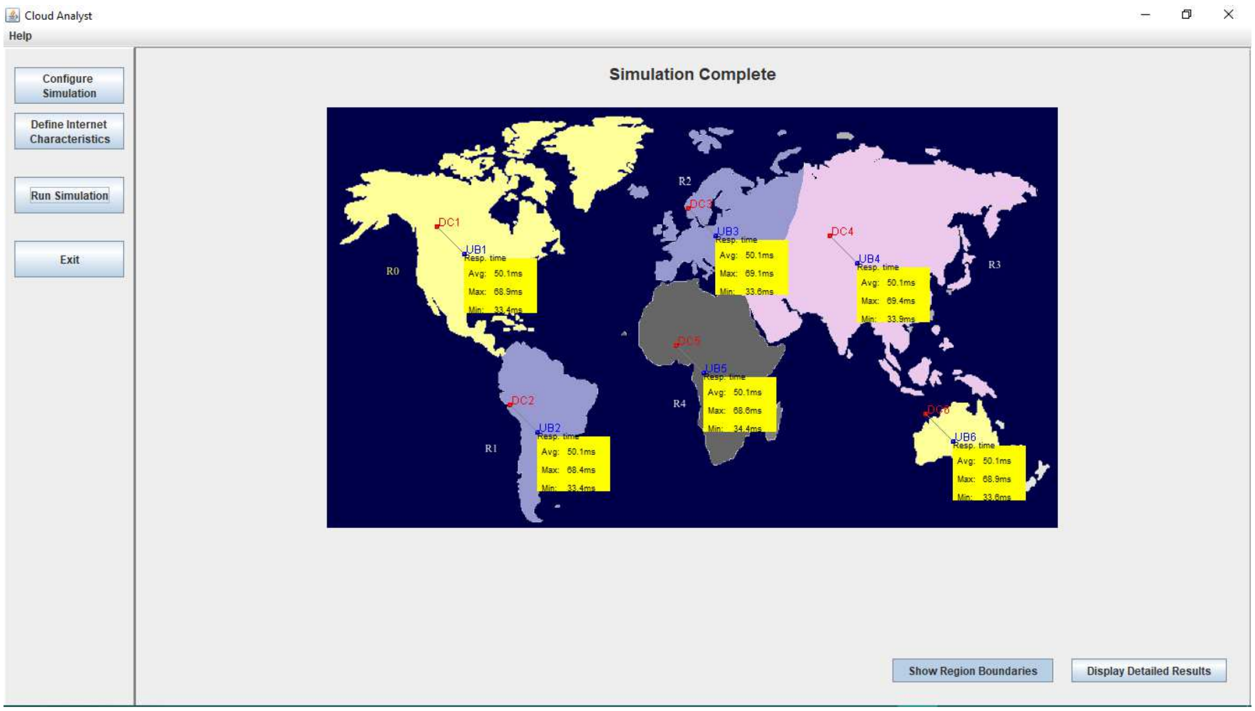

Figure 5: Case 1 for 150VMs simulation result

Table 5 shows the result from several cases using 150 VMs

Case	Overall Response Time	Data Center Processing Time
1	55.96	6.33
2	56.01	6.36
3	50.18	0.56
4	50.22	1.01
5	56.09	6.43
6	55.94	6.31

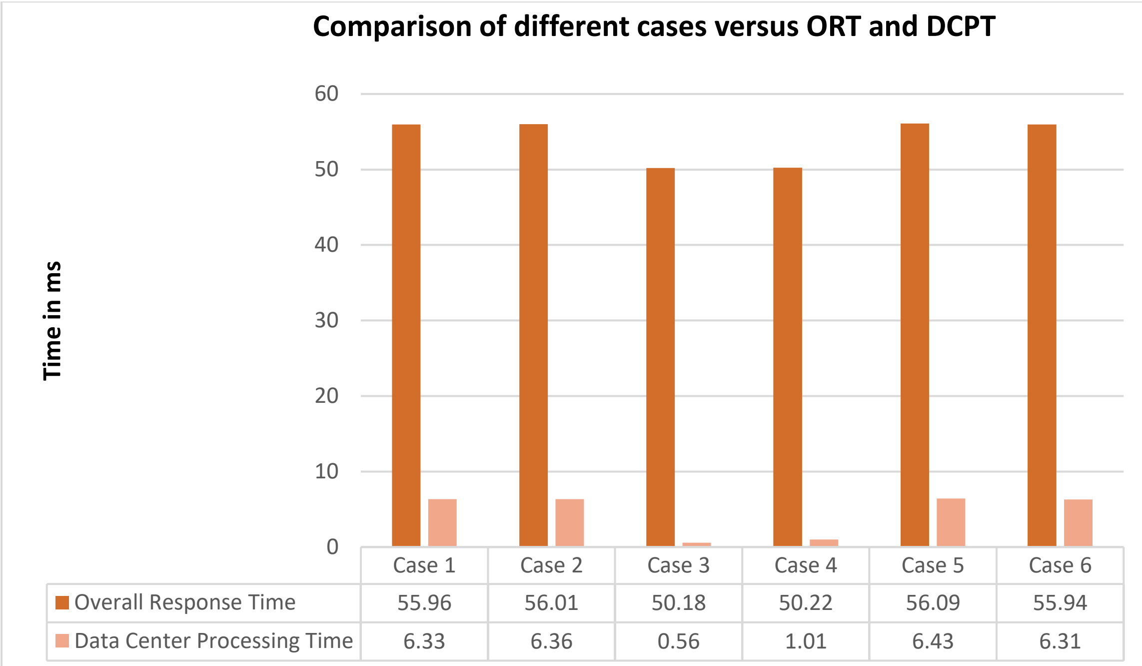

Figure 5: Comparison of cases versus ORT and DCPT

The results in Table 5 and Figure 5 reveal that the best algorithms are a blend of TLB Policy and ORTP Service Broker since the best RT service broker chooses the DCs with the lowest RT. The worst combination, on the other hand, is ESCELB with ORTP Broker, which has the slowest reaction time.

4.4 Comparing the outcomes of six cases for 50VMs, 100VMs and 150VMs

The study shows that the TLB Policy and ORTP Service Broker is the most effective algorithm mix to deploy since it has the best reaction time, which means less usage of infrastructure and resources, which saves both energy and money, as shown in tables 1-5 and Figures 1-5. The ESCELB and ORTP Service Broker algorithm combination, on the other hand, is the worst algorithm combination since it takes the longest time to respond, consumes the most power, and increases overall cost. Observation demonstrates that as the number of virtual machines increases, so does the overall response time, meaning increasing power consumption.

The mixture of TLB and RDLP is also very good, the response time is minimum and data center processing time is also good.

5.0 Conclusion and Future Work

In order to lower the energy costs of running cloud computing services, this study looked into the methodologies and algorithms used in Green Cloud Computing. Huge networks carrying services and data centers use a lot of energy, which results in a lot of CO2. As a result, the GCC concept has arisen and evolved in order to reduce the usage of electrical energy and global carbon dioxide emissions while simultaneously maintaining the services' long-term profitability, dependability, and efficiency. In the research, Cloud Analyst simulation toolkit was used to model six data centers in regular operation for six users in diverse locations during a 365-day period.

The results reveal that the TLB Policy and ORTP Service Broker combination is the most effective strategy since it has the fastest reaction time, which means it uses less infrastructure and resources, saving energy and money. On the other side, the ESCELB and ORTP Broker method is the worst algorithm combination since it takes the longest time to respond, uses more energy, and increases overall cost.

In general, this study advises users to choose the TLB Policy and ORTP Service Broker since they provide the fastest reaction time, implying less use of resources and equipment, and hence reduced costs. Businesses can save energy and money through Green Cloud computing.

Future work will still need to be carry out to find out if the use of another strategy can be more efficient than the one used in this paper.

DECLARATION: We declare no conflict of interest in this research work.

This project work is supported by Landmark University, Omu-Aran, Kwara State, Nigeria.

All abbreviation has been declared on first citing.

AUTHOR'S CONTRIBUTION: The project work was divided into sections, while some work on the background and introduction of the project, another works on the simulation of the several cases of the project, after gathering the results, other works on analyzing and discussion of the result and conclusion and recommendation were done together by the whole group.

REFERENCES

1. Yamini, R. (2012). Energy aware green task assignment algorithm in clouds. *Ianternational Journal for Research in Science and Advance Technology*, *1*(1), 23-29.2.

2. Aremu, D. R., & Moses, A. K. GRID, CLOUD, AND BIG DATA: TECHNOLOGIES OVERLAPS. *International Journal of Information Processing and Communication*.

3. Butler, B. (2013). PaaS Primer: What is platform as a service and why does it matter. *Network World*, *11*.

4. Yamini, R. (2012). Power management in cloud computing using green algorithm. In *IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM-2012)* (pp. 128-133). IEEE.

5. Raza, K., Patle, V. K., & Arya, S. (2012). A review on green computing for eco-friendly and sustainable it. *Journal of Computational Intelligence and Electronic Systems*, *1*(1), 3-16.

6. Sidhu, A. K., & Kinger, S. (2013). Analysis of load balancing techniques in cloud computing. *International Journal of computers & technology*, *4*(2), 737-741

7. Kansal, N. J., & Chana, I. (2012). Existing load balancing techniques in cloud computing: a systematic review. *Journal of Information Systems and Communication*, *3*(1), 87.

8. Amandeep, V. Y., & Mohammad, F. (2014). Different strategies for load balancing in cloud computing environment: a critical study. *International Journal of Scientific Research Engineering & Technology (IJSRET)*, *3*(1), 85-90.

9. Beloglazov, A., & Buyya, R. (2012). Managing overloaded hosts for dynamic consolidation of virtual machines in cloud data centers under quality of service constraints. *IEEE*

transactions on parallel and distributed systems, *24*(7), 1366-1379.

10. Ghomi, E. J., Rahmani, A. M., & Qader, N. N. (2017). Load-balancing algorithms in cloud computing: A survey. *Journal of Network and Computer Applications*, *88*, 50-71.

11. Milani, A. S., & Navimipour, N. J. (2016). Load balancing mechanisms and techniques in the cloud environments: Systematic literature review and future trends. *Journal of Network and Computer Applications*, *71*, 86-98.

12. Kalra, M., & Singh, S. (2015). A review of metaheuristic scheduling techniques in cloud computing. *Egyptian informatics journal*, *16*(3), 275-295.

13. Mesbahi, M., & Rahmani, A. M. (2016). Load balancing in cloud computing: a state of the art survey. *International Journal of Modern Education and Computer Science*, *8*(3), 64.

14. Kanakala, V. R., Reddy, V. K., & Karthik, K. (2015, March). Performance analysis of load balancing techniques in cloud computing environment. In *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)* (pp. 1-6). IEEE.

15. Shah, J. M., Kotecha, K., Pandya, S., Choksi, D. B., & Joshi, N. (2017, May). Load balancing in cloud computing: Methodological survey on different types of algorithm. In *2017 International Conference on Trends in Electronics and Informatics (ICEI)* (pp. 100-107). IEEE.

16. Neghabi, A. A., Navimipour, N. J., Hosseinzadeh, M., & Rezaee, A. (2018). Load balancing mechanisms in the software defined networks: a systematic and comprehensive review of the literature. *IEEE Access*, *6*, 14159-14178.

17. Walsh, W. E., Tesauro, G., Kephart, J. O., & Das, R. (2004, May). Utility functions in

autonomic systems. In *International Conference on Autonomic Computing, 2004. Proceedings.* (pp. 70-77). IEEE.

18. Kim, N., Cho, J., & Seo, E. (2014). Energy-credit scheduler: an energy-aware virtual machine scheduler for cloud systems. *Future Generation Computer Systems*, *32*, 128-137.

19. Torrens, J. I., Mehta, D., Zavrel, V., Grimes, D., Scherer, T., Birke, R., & Pesch, D. (2016, April). Integrated Energy Efficient Data Centre Management for Green Cloud Computing. In *Proceedings of the Proceedings of the 4th International Conference on Cloud Computing and Services Science, Rome, Italy* (pp. 23-25).

20. Lee, H. M., Jeong, Y. S., & Jang, H. J. (2014). Performance analysis based resource allocation for green cloud computing. *The Journal of Supercomputing*, *69*(3), 1013-1026.

21. Abiodun Kazeem Moses, Awotunde Joseph Bamidele, Ogundokun Roseline Oluwaseun, Sanjay Misra & Adeniyi Abidemi Emmanuel (2021) Applicability of MMRR load balancing algorithm in cloud computing, International Journal of Computer Mathematics: Computer Systems Theory, 6:1, 7-20, DOI: 10.1080/23799927.2020.1854864