

A REVIEW ON EFFECTIVE UTILIZATION OF COMPUTATIONAL RESOURCES USING CLOUDSIM

G. Anusooya¹, Md Abdul Quadir², J. Christy Jackson³, J. Prassanna⁴, S. Bharathi Raja⁵, Sakkaravarthi Ramanathan⁶

¹School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India.

²School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India.

³School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India.

⁴School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India.

⁵School of Computer Science and Engineering, Vellore Institute of Technology, Chennai, India.

⁶Department of Computer Science, Cegep Gaspesie, Canada.

Received: 20.04.2020

Revised: 22.05.2020

Accepted: 17.06.2020

Abstract

Data Center is a very important part of cloud computing which involves different database types that store and run all types of information. Data Center containing network, server, etc. absorbs more resources, more electricity, and more carbon emissions. Workload management is a process of directing the proper distribution of workloads to provide applications with more appropriate performance. A hierarchical approach is also presented. This approach includes 5 algorithms. In this survey, multiple techniques and algorithms are presented to minimize the energy, and power consumption in the cloud.

Keywords: Cloud Computing, Data Center, Energy Saving, VMs Migration, Energy Efficient Techniques, Cloud Analyst.

© 2020 by Advance Scientific Research. This is an open-access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>)
DOI: <http://dx.doi.org/10.31838/jcr.07.13.177>

INTRODUCTION

Since a data center expends 30% to 50% of operating expenses in electricity. Therefore, it is necessary to reduce energy and power utilization and carbon emissions. Cloud computing provides some services over the internet. The cloud may be public, private, hybrid and community. A data center can be defined as a storage which is used to organize, store, and access large amount of data. Mostly it is useful for businesses and other organizations. A business or organization typically heavily depends on the application, services, and data contained in a data center. Virtualization is a technique, in which different operating system can be installed on hardware. They are completely independent and separated from each other. Virtualization is a method in which machine is shared by multiple users and customers. There are different types of virtualization: i) Server Virtualization ii) Client & Desktop Virtualization iii) Services and Applications Virtualization iv) Network Virtualization v) Storage Virtualization. Virtualization includes specific advancements, such as consolidation and resource utilization. Green computing is non-polluting and ecological utilization of computer resources. Load balancing is a method in which workload is distributed among various computer [1]. A load balancer distributes client requests or network load efficiently across multiple backend servers. Workload Management [2] is a challenging task and to handle this task, a literature survey is Done in which 5 algorithms, Round robin, Equally Spread Current Execution, Throttled, HoneyBee, and AntColony are used. In this survey, we have used CloudAnalyst simulation tool for comparing performance of various algorithms to check overall response time. CloudAnalyst tool looks like uncomplicated as it has graphic environment with which it looks simple to apply.

It also happens to have a simulation standard that is much greater than a tool set. There are several tools available for modeling environment and could be apply to test the output of applications on web. Cloud Observer distinguishes simulation setting up system exercise and encourages the modeler to concentrate on the criteria used for simulation purposes instead of only programming techniques. In greatly little time, it also helps simulation by changing variables readily and immediately.

The cloud analyst's graphical user interface simulation tests allow the findings to be analyzed more explicitly and more proficiently, as well as helping to emphasize any issues with simulation accuracy and efficiency.

LITERATURE SURVEY

[3] The aim of the author is to reduce the amount of carbon emission using effective utilization of resources. The work is carried out by balancing the load among various resources evenly. It also allocates the work based on the content of the request sent either as read only or read write operation. Effective and appropriate resources are also the major contribution of the work. By assessing the nature of the request the work is allocation to its appropriate available resources. Due to this they have reduced the amount of carbon emission used in a traditional system and their proposed water shower model.

[4] The aim of the author is to use an efficient way of power consumption model named balanced exothermic dynamic voltage carbon scaling which is used for analyzing the power of the resources and according the job is allocated. They also use CPTS, circular peak time services to monitor the entire power usage and carbon emission by this complete monitoring of resources the author has proved reduced carbon emission and power consumption using that model.

[5] The aim of the author in this survey is to examine energy for the efficient utilization of data centers and resources. This survey represent some virtualization and data center- based methods. It also present a revision of vivacity efficiency technique in cloud computing on the source of presentation and energy saving. The reason of this survey is to discover up to date for energy and presentation management.

[6] In this survey, a hierarchical approach, EnergyCloud for the workload of data center is presented. Two algorithms are used in this survey which is used for workload assignment and migration. In this survey architecture is also presented for sharing workload. In this architecture two scenarios are defined, EnergyCloud framework for two interconnected data center and

EnergyCloud framework for four interconnected data center. The purpose of this survey is to distribute the workload among different data centers.

[7] Authors are studying the energy-efficient arrangement of responsibilities in this survey, where the implementation time of the job is unknown. To describe the effectiveness of unlike frequencies, they characterize a novel task model to explain the responsibilities and the energy consumption ratio. They show that the job allotment is related to and additional difficult than the inconsistent size bin packing difficulty. Then, to assign jobs, they represent two successful heuristics. They also plan an algorithm for confined job migration to recover presentation when a job is complete. Ultimately, this survey presents a model framework for evaluating their approaches, which achieves better energy-saving performance.

[8] In this survey, a project named DATAZERO is represented. The aim of this survey is to find the solution about the plan and operations of data center. The project's key originality is to suggest a concession system between IT and power management, which seeks to seek a tradeoff between the priorities and limitations of both sides instead of trying to explain a question of universal optimization. They suggest reliable electrical and IT models to make this concession probable: they outline the need for identification of electrical sources.

[9] In this survey they used technique dynamic voltage frequency scaling for minimizing data consumption and performed cloudsim toolkit using real cloud traces, and considered DVFS would be necessary when mapping virtual machines to maintain quality of services. Their result demonstrate that including DVFS awareness is workload management provides substantial energy saving till 41.62% for scenarios below workload conditions.

RELATED WORK

CloudAnalyst[10] tool is based on cloudsim works with java packages, it is completely based GUI for giving graphical output. It is very simple to use with a scale of visualization ability. Its graphical result helps user to analyze results more readily and most effectively. Characteristics of CloudAnalyst are:

- i. Easy to apply- As in the case of javaPackage, CloudAnalyst is very simple to apply in setting simulation environment, we must to doubleclick on icons. User can change the variables as many times and run simulation easily, it is very simple to setup configuration of CloudAnalyst. Only the thing is to do by user is to enter number of datacenter as required and number of VMs for each datacenter ,and user have to set the regions as required.
- ii. Defining simulation functionality with a large point of configuration and versatility- This tool may give degree of configuration is most important part possibly. Web application upto on many variables and most often the amount of that variables require to be presumed. So it is necessary to input and modify that variable readily and rapidly.
- iii. Graphic based result - CloudAnalyst is able to show output in form of table and charts which helps to analyze result easily and quickly. Graphical output helps to compare the result effectively. It can easily compare overall response time, Avg, minimum, maximum, and cost readily.
- iv. Repetitions of experiments - CloudAnalyst has important features to repeat experiment as many times as user want to repeat. If user is applying an experiment having some variable or input then on simulation it shows some result, so if users apply same experiment with same variable, on simulation same result will be obtained.
- v. Capability to save the output - On simulating CloudAnalyst when user apply any experiment, user can save that experiment file and also can save the output file in pdf format. These files can be saved on computer or can save in

any pendrives, which helps user to use that files for longer time.

MAIN COMPONENTS OF CLOUDANALYST

The following part defined about the main component of Cloud Analyst and fig1. Describes the system of Cloud Analyst.

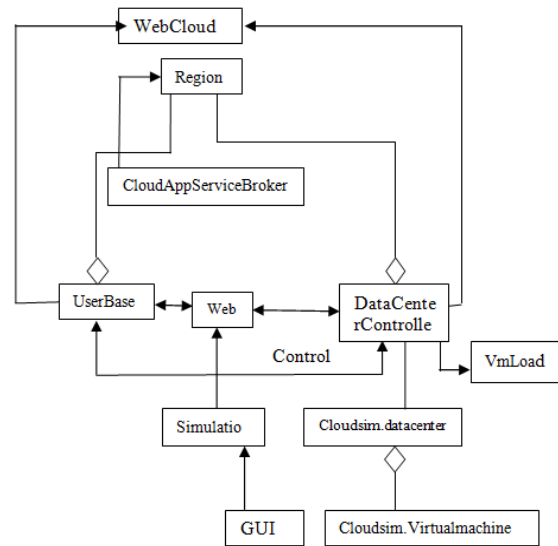


Fig. 1: Cloud Analyst System

a) Regions - The world's main continent are divided into 6 Regions in the CloudAnalyst. All of these areas belong to the further key bodies like userbases and datacenters. In the CloudAnalyst this geographical categorization is very helpful for supporting the degree of its easy and simple models for simulations and the output are gained.

b) Web - The CloudAnalyst web is a proposal for the world web which implements only that characteristics which are important for simulation. Cloud analyst also controls the duplicates of web traffic, which looks scattering on all side of world along organizing suitable quantity data transportation delay and transfer impotency. Convenient bandwidth and transfer impotency between six areas can be configured by users.

c) CloudAnalyst Service Broker - Service Broker controls all obstruction which is routing among userbases and datacenters. Service broker also decides about maintenance of datacenters from one and all userbases. Separate routing strategies are implemented by three kinds of service broker policy which are provided by CloudAnalyst tool at present. Closest datacenter, Optimize Response Time, Reconfigure Dynamically Load Balancing are three kind of server broker policy.

d) UserBases - Group of users participated in CloudAnalyst are taken as single unit are called as UserBases. Individual userBase can express millions of users but configured like an individual unit.

e) Data Center Manager - DCMs a variety of variables summarizing the site's status: potential variables include total resource, operational capacity, energy costs, CO2 emissions. After receiving this information from other pages, data center should be allocated to a new application or VM in compliance with the stated goals and very that the workload is adequately spread between the different sites and cause the relocation of applications where necessary.

f) VM Load Balancer - Data Center make use of VM LoadBalancer., to control each VM assigned for refining of forthcoming cloudlet. At present, CloudAnalyst tool has 3 VM LoadBalancer for three already available loadbalancing

algorithms for implementation, but user can manage and add their own load balancer for implementing other added load balancing algorithms. As per required, that policies can be chosen by the simulator. In this survey, we have used three VM LoadBalancer, roundrobin, throttled, and active monitoring.

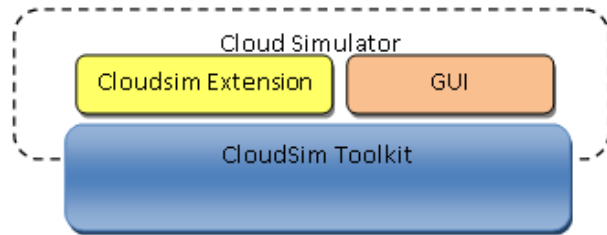


Fig. 2: Cloud Analyst Architecture

ALGORITHM

In this survey, CloudAnalyst is used for comparing different through distributing load on each data center, by using java packages. There are five algorithms are used for comparing their performance on basis of their response time and data processing time. By using VM LoadBalancer [11], Data Center balanced the load request on all VMs. Algorithms are described as follows:

Round Robin Load Balancing Algorithm

Round Robin Load balancing algorithm is very simple and easy procedure which is based on the time quantum and applies round robin.

```
Rrlb()
{
[initialize]Time_Quantum as TQ=10
Repeat steps until
(actual_request_list).size==NULL
{
Perform[actual_request] till TQ
Performed_list [requests] performance[Actual_request]
actual_requestrequest_list[next_list]
}
VM[performed_request_list] performed_list [requests]
Go to step 4
```

Throttled Load Balancing Algorithm

Steps of procedure are as follows-

1. At first, in the starting of procedure, check all the VMs should be present. Throttled VM LoadBalancer managed list of virtual machines, also checked that whether VM is occupied or idle.
2. DCM collects new cloudlet.
3. DCM inquired VM load balancer about the upcoming allocation.
4. VM load balancer examine the table of virtual machine from top to bottom until present exist VM is assigned. If assigned:
 - a) Virtual machine id is send back to the DCM.
 - b) Then, DCM moves that cloudlet to virtual machine which identified by particular id.
 - c) DCM informs load balancer recondition the allotment table appropriately. If not assigned:
 - a) Throttled VM load balancer give back -1.
 - b) The DCM line up the cloudlet
 - c) When virtual machine complete all the cloudlet process and also DCM has been sent the retaliation , then DCM informed load balancer to execute de-assignment of particular VM.
 - d) DCM inspect about the stand by cloudlet in line, if there exist, repeat steps from 3.
5. Begins from step2.

Equally Spread Current Execution LoadBalancing Algorithm

Steps of procedure are as follows:

1. At first, in the starting of procedure, check all the VMs should be present. Active VM LoadBalancer managed list of virtual machines, also checked that whether VM is occupied or idle.
2. DCM collects new cloudlet.
3. DCM inquired VM load balancer about the upcoming allocation
4. Applying changed Active Monitoring algorithm. This active loadbalancer observe the accessible VM with smallest vertex.
5. If the least vertex of one or more VM are identical, then earliest recognized VM would be allotted.
6. Then , after recognition, Active VM load balancer sent back the VM id to the DCM.
7. After that cloudlet are transferred by the DCM to the allocated VMs , correlating id identifies the virtual machine.
8. DCM informed Active load balancer about the current assignment of VM
9. Then Active monitoring load balancer recondition the allotment table appropriately.
10. When virtual machine complete all the cloudlet process and also DCM has been sent the retaliation , then DCM informed active load balancer to execute de-assignment of particular VM.
11. At the end, then Active monitoring load balancer recondition the allotment table appropriately which decrease the allotment count by one for that certain VM.
12. Begin from step4.

Honey Bee Load Balancing Algorithm

As per user request, requests are dynamically in load balancing. Here, the requests are converted into groups and each virtual machine manages a procedure. After cleaning, gain is calculated. If there is more gain, server will stand, else low gain activate a return. Each vertex manages a distinct line. After determining threshold value,load is transferred to unplanned allocated under underloaded virtual machine. Due to this, determination of virtual machine having high presentation is not possible. Initially calculating

Capacity, particular VM would be allocated based on highest value of capacity. Steps of algorithm are as follows:

1. In the beginning of algorithm, set number of job.
2. Then put number of virtual machine and determine capacity.
3. At the beginning, on each virtual machine put the load equals to null.
4. Select virtual machine which is having highest throughput, and send first job to that virtual machine.
5. Checks that at present VM load is greater than threshold value.
6. If alright, choose the virtual machine which have highest throughput and examine the threshold value of load, if it is then allocate to that virtual machine.
7. If not, then choose another virtual machine with succeeding highest throughput and check load is less than threshold value.
8. If virtual machine received all the jobs, go to steps 10.
9. Otherwise go to step4.
10. Stop

Ant Colony Load Balancing Algorithm

Steps for this algorithm are as follows:

1. In the beginning of algorithm, set pheromone for existing nodes.
2. Set the ants and set them on existing virtual machines randomly.

3. Calculate moving probability of ants p as per gain matrix, and select following node.
4. Limited pheromone would be upgraded if ant p finishes the searching cycle, and if not, go back to step 3.
5. Universal pheromone would be upgraded, if all ants finishes the searching cycle, and if not go back to step2.
6. Check if any virtual machine is remained to be allocated in the list and if yes, go back to step1.
7. Otherwise stop.

IMPLEMENTATION

The load balancing algorithms deployed shall be evaluated in a CloudAnalyst. CloudAnalyst contains the software Cloudsim. Cloudsim is a Java-based, application library. By integrating with

the JDK this library can be used directly to compile and execute the code.

Steps for Implementation:

1. Download CloudAnalyst package
2. Download eclipse and install on computer
3. Import cloud analyst package on eclipse
4. Now run the CloudAnalyst, it will automatically open the GUI for CloudAnalyst.
5. Implement as many as algorithm and import it on CloudAnalyst and run simulation, by setting up the entire configuration.
6. Set userbase as required and data centers. Set as many as VM for each data center as per user requirement.



Fig. 3: GUI of Cloud Analyst

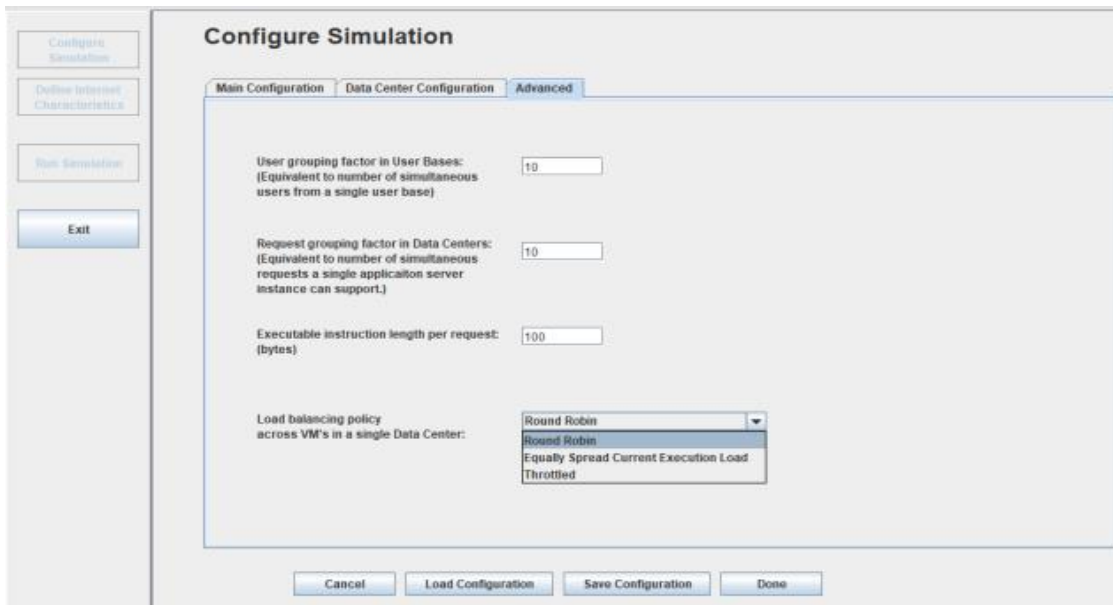


Fig. 4: Applying Algorithm in Cloud Analyst

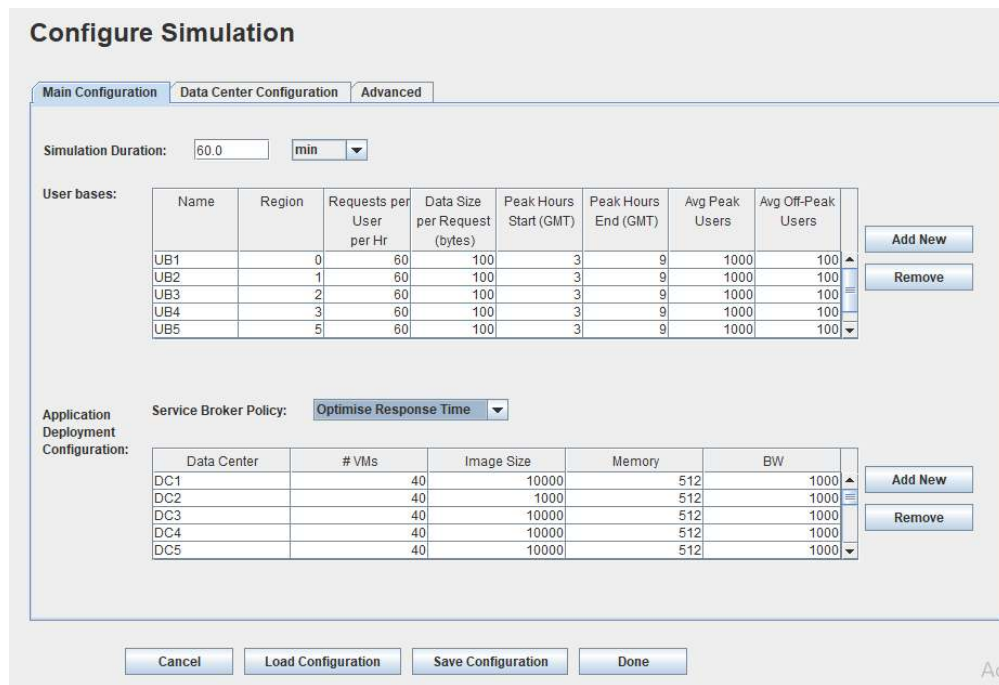


Fig. 5: Configuration of Cloud Analyst

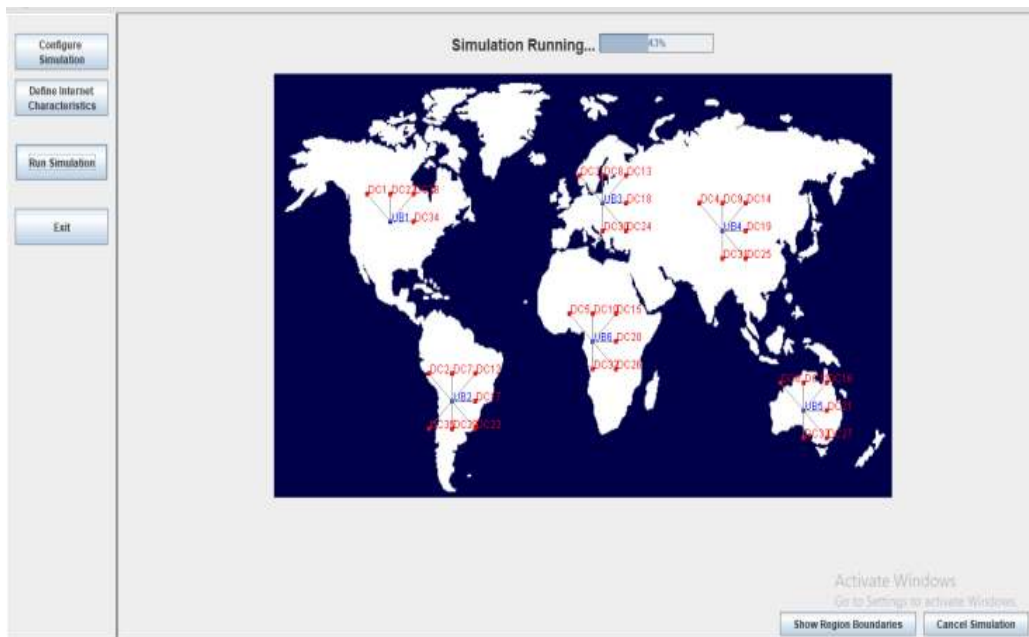


Fig. 6: Running Simulation

RESULT ANALYSIS

In this survey, CloudAnalyst is used for comparing the performance of various load balancing algorithm by taking different no of data centers and different of virtual machine for each data center. Cloudanalyst tool helps to examine various loadbalancing algorithms on distributing load on virtual machine of each data center. By taking various userbases this environment is simulated and each simulation is run for 60 minute. For comparison, Avg peak users are taken as 10000 and Avg -off peak user as 100 for each userbase. For simulation, optimize response time is used as service broker policy.

When Round Robin Load Balancing is applied :

For distributing load on each VM, Round Robin method is used for every VM. For comparison, 35 data centers and 40 VMs for each data center are taken for Round Robin algorithm. Result is represented in table 1.

Table 1: Output of Round Robin LoadBalancing Algorithm

Variable	Avg(ms)	Min(ms)	Max(ms)
Overall Response Time	50.54	40.16	63.12
Datacenter Processing Time	1.06	0.00	1.76

The output obtained for the cost variable after simulation is 140.27\$.

When Equally Spread Current Execution Algorithm is applied:

For distributing load on each VM, Equally spread method is used for every VM. For comparison, 30 data centers and 35 VMs for each data center are taken for Equally Spread algorithm. Result is represented in table 2.

Table 2: Result of Equally Spread Load Balancing Algorithm

VVariable	Avg(ms)	Min(ms)	Max(ms)
Overall Response Time	50.30	41.10	62.11
Datacenter Processing Time	0.97	0.00	1.64

The result obtained for the cost variable after simulation is 105.24\$

When Throttled Algorithm is applied:

For distributing load on each VM, Throttled method is used for every VM. For comparison, 40 data centers and 45 VMs for each data center are taken for Throttled algorithm. Result is represented in table 3.

Table 3: Output of Throttled Algorithm

Variable	Avg(ms)	Min(ms)	Max(ms)
overall Response Time	50.42	41.29	61.61
Datacenter Processing Time	0.71	0.03	1.38

The output obtained for the cost variable after simulation is 79.85\$

When HoneyBee Algorithm is applied:

For distributing load on each VM, HoneyBee method is used for every VM. For comparison, 45 data centers and 50 VMs for each data center are taken for HoneyBee algorithm. Result is represented in table 4.

Table 4: Output of HoneyBee Algorithm

Variable	Avg(ms)	Min(ms)	Max(ms)
Overall Response Time	50.95	40.26	62.76
Datacenter Processing Time	1.23	0.07	2.02

The output obtained for the cost variable after simulation is 226.18\$.

When AntColony Optimization Algorithm is applied:

For distributing load on each VM, Antcolony method is used for every VM. For comparison, 50 data centers and 55 VMs for each data center are taken for AntColony algorithm. Result is represented in table 4.

Table 5: Output of AntColony Algorithm

Variable	Avg(ms)	Min(ms)	Max(ms)
Overall Response Time	51.05	38.60	63.06
Datacenter Processing Time	1.33	0.14	2.15

The output obtained for the cost variable after simulation is 276.37\$.

On Comparing different algorithm with different variables, the graph of Total Response Time and Datacenter Accessing Time obtained are represented in Fig 7. and Fig 8.

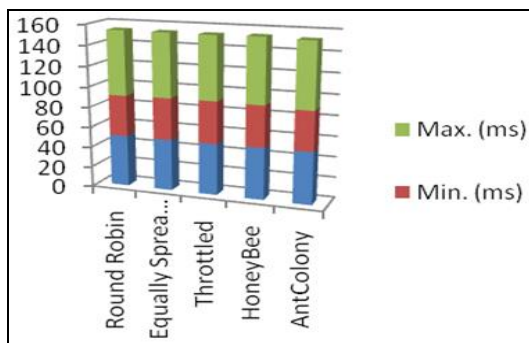


Fig. 7: Overall Response Time of Load Balancing Algorithms

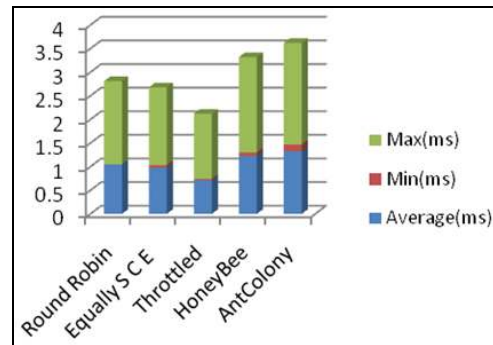


Fig. 8: Datacenter Accessing Time of Load Balancing procedures

CONCLUSION

This literature survey focused on energy and power saving in clouds and workload management in data centers. For energy and power saving five methods are presented- RoundRobin, Throttled, Equally Spread Current Execution, HoneyBee, Antcolony Optimization. After comparing various algorithm in CloudAnalyst tool, the output obtained by taking different variable in each simulation of different algorithms. The response time of equally spread current execution algorithm is fine for 30 datacenter and 35 VMs in comparison of other four algorithms. The datacenter processing time of Throttled algorithm is good as compare to other four algorithms. As compare to other four algorithms, the cost of throttled Algorithm is least. According to the result, Equally Spread Current Execution proves to be good in response time and Throttled Algorithm proves to be good in both cost and data center processing time.

REFERENCES

1. Prassanna J & Neelanarayanan V (2019), Threshold Based Multi-Objective Memetic Optimized Round Robin Scheduling for Resource Efficient Load Balancing in Cloud, Mobile Network and Applications, Volume 24, Issue 4, pp 1214–1225.
2. Prassanna J & Neelanarayanan V (2019), Adaptive Regressive Holt–Winters Workload Prediction and Firefly Optimized Lottery Scheduling for Load Balancing in Cloud, Wireless Network.
3. Anusooya, G.; Vijayakumar, V. Reduced carbon emission and optimized power consumption technique using container over virtual machine. Wireless Networks 2019, pp. 1–19.
4. Anusooya, G.; Vijayakumar, V.; Narayanan, V.N. Reducing the carbon emission by early prediction of peak time load in a data center. Journal of Intelligent & Fuzzy Systems 2019, 36, 4341–4348.
5. Shaheen, Q., Shiraz, M., Khan, S., Majeed, R., Guizani, M., Khan, N., & Aseere, A.M. (2018). Towards Energy Saving in Computational Clouds: Taxonomy, Review, and Open Challenges. IEEE Access, 6, 29407-29418.
6. Forestiero, A., Mastroianni, C., Meo, M., Papuzzo, G., & Sheikhalishahi, M. (2016). Hierarchical approach for efficient workload management in geo-distributed data centers. IEEE Transactions on Green Communications and Networking, 1(1), 97-111.
7. Wang, S., Qian, Z., Yuan, J., & You, I. (2017). A DVFS based energy-efficient tasks scheduling in a data center. IEEE Access, 5, 13090-13102.
8. Pierson, J.M., Baudic, G., Caux, S., Celik, B., Da Costa, G., Grange, L., & Rehn-Sonigo, V. (2019). DATAZERO: Datacenter With Zero Emission and Robust Management Using Renewable Energy. IEEE Access, 7, 103209103230.
9. Arroba, P., Moya, J.M., Ayala, J.L., & Buyya, R. (2017). Dynamic Voltage and Frequency Scaling-aware dynamic consolidation of virtual machines for energy efficient cloud

- data centers. *Concurrency and Computation: Practice and Experience*, 29(10), e4067.
10. B. Wickremasinghe, R.N. Calheiros and R. Buyya (2010), "CloudAnalyst: A CloudSim-Based Visual Modeller for Analysing Cloud Computing Environments and Applications", 24th IEEE International Conference on Advanced Information Networking and Applications, Perth, WA, 2010, pp. 446-452.
 11. Prassanna J., Jadhav P.A., Neelanarayanan V. (2016) Towards an Analysis of Load Balancing Algorithms to Enhance Efficient Management of Cloud Data Centres. Proceedings of the 3rd International Symposium on Big Data and Cloud Computing Challenges (ISBCC - 16'). Smart Innovation, Systems and Technologies, vol 49. Springer, Cham.