International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249-8958 (Online), Volume-8 Issue-3, February 2019

Performance Evaluation in Implementing a Multi-Layer Job Scheduling Approach with Energy Efficient Resource Utilization Over a Cloud

Vinod Kumar Saroha, Sanjeev Rana

***Abstract:** A cloud computing provides the platform where numerous users and companies are connected for accessing different types of services such as software, applications, platforms and infrastructure. This technique utilizes various online web based processing structures. There is one major issue regarding the energy consumption and dissipation of cloud server during processing of their routine tasks. In this research work, our chief focus is to investigate and reduce energy consumption with enhanced scheduling approach based on multi-layer architecture. Here, we propose and implement a multi-layer scheduling approach and request the load balancer for managing multiple job queues along with effective resources utilization over cloud network. This is performed by the creation of client server database on the cloud where the servers are categorized as highest, intermediate and lowest priority server on the basis of the configurational parameters of processing speed, RAM and time; while the client requests are classified on the basis of urgency (processing need) and assigned the priorities likewise. A high priority job request is executed by higher configuration server while the lower tasks are accomplished by the lower configuration servers; that helps in energy saving..We evaluate our work with various performance parameters viz. energy, network (processor) utilization and response (processing) time to get optimal results. The evaluation work involves different scheduling and load balancing cloud computing algorithms viz Round Robin procedure, Minimum Completion Time (MCT) algorithm and Opportunistic Load Balancing (OLB) etc.; for efficiently utilizing the resources. The comparative study of the proposed algorithmic approach outperforms the earlier ones and yields better energy efficiency.*

***Keywords:** Resource scheduling, Efficiency of energy, Throughput, Response time, Processor Utilization.*

I.INTRODUCTION

Cloud Computing has formed the vital platform for various computational web based applications and has hence evolved itself since years. It provides us the procedures for sharing of variety of resources: storage, processing and data flow for different types of applications. Job scheduling provides a thrust with help of effective algorithmic implementation and is the backbone of cloud computing area. It further leads to the energy efficient flow of information in the cloud processing and aids in further researches. The services of the cloud computing caters to large scale computing applications by utilizing various storage resources.

Manuscript published on 28 February 2019.
***** Correspondence Author (s)
Vinod Kumar Saroha, PhD CSE (Research Scholar) Department of CSE, MMDU, Mullana, Ambala, Haryana, India.
Dr. Sanjeev Rana, Professor, Department of CSE, MMDU, Mullana, Ambala, Haryana, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license http://creativecommons.org/licenses/by-nc-nd/4.0/

Hence it is very economical and efficient as the users have not to incur expenses on its maintenance, storage or infrastructure costs. It is very similar for the users availing its services and paying for the same; like in water, gas, electricity or any other utility service. Scheduling algorithms are used for dispatching user tasks or jobs to a particular resource or data. Job scheduling is a very tedious task as there are limited number of resources with varied efficiency and potential. A cloud computing is the technique with which numerous users and companies are connected for accessing different types of services such as software, applications, platforms and infrastructure. However, the energy use and its optimal flow remains a big consideration while operating routine jobs in a cloud. Here, our main focus is on to investigate and reduce energy consumption with enhanced scheduling approach based on multi-layer architecture. In this work we propose and implement a multi-layer scheduling approach and request load balancer for managing multiple job queues along with effective resource utilization over cloud network.

II.IMPLEMENTATION WORK:

At the onset, database of client-server is established on cloud. The servers are classified in accordance to their speed, processing time and internal memory. While the clients are categorized for their job requests and are classified into three priority types; viz. highly prioritised, of middle priority and lowest priority depending upon the urgency of the processing need for the job requests. The servers on the basis of the configuration; are categorized as lower, intermediate and higher priority servers. The job requests of the client that take longer duration of time will require the configurational server with highest priority.
In this study we have presented a multilayering job scheduling based approach on basis of requirements for both the client and the server. A job request with high priority is executed by higher configuration server while the lower job requests are entertained by the lower configuration servers; that helps in energy optimisation. When; the higher server has finished up with its assigned jobs; then job requests from lower server are transferred to the server with higher priority; which results in balancing of load on the cloud and effective resource allocation. The overall process has been best explained with help of example illustrated herewith; ie. the servers will execute the tasks assigned in given stipulated time. This fast execution helps the servers to liberate themselves quickly and helps in better energy saving.

© Copyright: All rights reserved.

The proposed work after implementation is best illustrated with the help of diagrams.
The following figures will elaborate the proposed work of our implementation.
The client login screenshot is shown in figure 1.

Figure 1: The client login screen

Figure 2. shows the processing time required for each job.

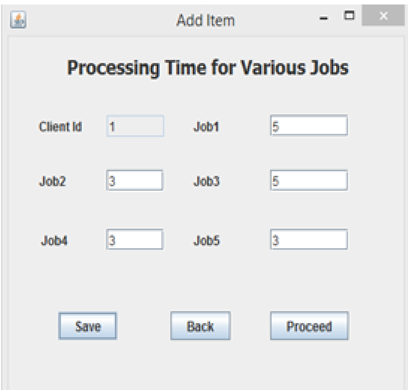

Figure 2: Processing time each job for client id 1

III.EVALUATION CRITERIA:

There are different scheduling and load balancing cloud computing algorithms. The basic motive of different algorithms is to efficiently utilize the cloud computing resources. There exists different types of evaluation parameters with which we can determine whether an algorithm is optimal or not. These evaluation parameters are:

1. CPU Utilization: Prime objective of any evaluation parameter is to ascertain CPU should be kept busy.

2. Throughput: Another evaluation parameter is throughput. It determines numbers of jobs (processes) are completed in the given unit of time. We can measure throughput per hour basis for lengthy processes (jobs) and per second basis for processes (jobs).

3. Response time: It may be defined as the time period with respect to making a request till the delivery of response. The response time measures the time of processing submitted jobs.

We want both CPU utilization and server throughput be highest, and response time to be lowest.
Some optimization criteria are as follows:

● Maximum CPU utilization
● Maximum Throughput
● Minimum Response time

We can determine performance of any system using CPU utilization factor. Utilization is basically the ratio of Requirement and Capacity. This can be as elucidated: Let's take illustration. Consider a water bottle of capacity **1000** ml (C)
The bottle is half full i.e. **500** ml
Requirement to fill the bottle is additional **500** ml (R)
Hence, the utilization will be U= R divided C
= **500** / **1000** * **100**
= **0.5** * **100** = **50** %

Hence, if the bottle is half full the utilization percentage of the bottle is **50 %**
A easy method to evaluate CPU utilization is by using
U=R/C, where
U= Utilization
R= Requirements which is the BUSY TIME
C= Capacity which is simple terms is BUSY TIME + IDLE TIME
In our case CPU Utilization approaches to 100% in best case after performing scheduling & load balancing in cloud server.

CPU Utilization:
No of jobs arrived / maximum No of jobs * 100 = 5/5 * 100=100%
In worst case utilization is below 100%.

Response time is frequently measured as units of **"seconds / request" OR "seconds / transaction"**.

Throughput determines numbers of jobs (processes) are completed in the given unit of time. We can measure throughput per hour basis for lengthy processes (jobs) and per second basis for processes (jobs). It is usually calculated in terms of **"requests / second" or "transactions / second"**.

Let us take the example of general store. The cashier in normal case takes two minutes for processing of client works. Now let there are ten clients then we can compute both the response time, throughput as follows.

- The response time is total time required for processing all the clients divided by total clients as: **= 20 mins / 10 = 2 mins / client.**
- The processing time of first client is at zero minute, second cli3nt at two minutes and last client at twentieth minute i.e., 10 client / 20 mins = **.5 clients / minute**

Throughput = 1 / Response Time
In our case response time is 2 seconds / request for server 1 (8/4).
In our case response time is 2.5 seconds / request for server 2 (8/3.2).
In our case response time is 3.2 seconds / request for server 3 (8/2.5).
In our case response time is 3.8 seconds / request for server 4 (8/2.1).

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

In our case response time is 4.2 seconds / request for server 5 (8/1.9).

In our case throughput is 0.5 request/seconds for server 1 (1/2).

In our case throughput is 0.4 request/seconds for server 2 (1/2.5).

In our case throughput is 0.3 request/seconds for server 3 (1/3.2).

In our case throughput is 0.26 request/seconds for server 4 (1/3.8).

In our case throughput is 0.23 request/seconds for server 5 (1/4.2)

Calculating the energy efficiency:

We can compute energy consumption of any electronics device by the product of Power (watt) of device with the number of working hours of the device. The unit used for energy consumption is watt-hour. For instance if a television takes 120 watt and used four hours daily then we can compute energy consumption of electronics device (television) by multiplying these factors as:

120 watts x 4 hours = 480 watt-hours per day

Convert to Kilowatts :

Next step is to convert watt-hour into kilo watt hour of energy consumption of television. One kilo watt is 1000 watt. Hence kilo watt hour is obtained by dividing the watt-hour by 1000 as:

480 watt-hours per day / 1000 = 0.480 kWh per day

Monthly Usage of electronics device:

Next step is to obtain monthly usage of energy by the television. For 30 days month the kilo watt hour per month is computed by multiplying with 30 as:

0.480 kilo watt-hours per day X 30 days = 14.40 kWh per month

Computing the Cost:

Now, we compute the monthly cost of electronics device. Let cost of one kilo watt hour is Rs 3.50. Therefore the monthly cost of television is obtained by multiplying with Rs 3.50 as:

14.40 kWh per month X 3.50 per kWh = Rs 50.40 per month

We can compute energy consumption of personal computer by multiplying its usage in the month with the number of watts it taken which is approximately 100 watts.

In our case **energy efficiency** is better as lower utilization servers jobs are transferred to higher utilization servers. Now the lower utilization servers are closed thereby saving lot of energy.

IV.RESULT & ANALYSIS:

We can compare result of our proposed work with the result of following three scheduling algorithms:

- The Round Robin (RR) algorithm of scheduling transfers the given process in the round robin fashion to the virtual machines available for distribution.
- The Minimum Completion Time (MCT) algorithm of scheduling the job transfers the given process with minimum time of completion to the virtual machines available for distribution.
- The Opportunistic Load Balance (OLB) algorithm of scheduling transfers the given processes for processing to virtual machines with the minimum load with respect to other virtual machines.

Table 1 and Figure 3 below show the comparative chart of old approaches and our proposed scheduling algorithm for the throughput metrics shown as:

Table 1: The comparative chart of throughput metrics.

No. of vms	Round Robin	Minimum Completion Time	Opportunistic Load Bal	Proposed Algo
5	95	100	100	100
10	84	94	90	98
15	74	85	82	94
20	62	75	73	86
25	55	65	62	78
30	45	57	55	65

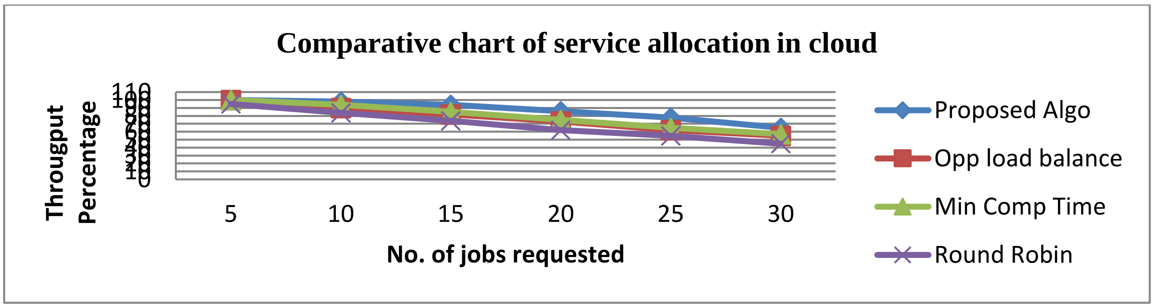

Figure 3: The comparative chart of old approaches and our proposed scheduling procedure for the throughput metrics.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

Table 2 & Figure 4 below show the comparison table of old techniques and our proposed scheduling algorithm for the average response time metrics.

Table 2: The comparison table of average response time metrics.

Algorithm	Avg Response Time
Round Robin	365.52
Min Comp Time	364.85
Opp Load Bal	362.67
Proposed	359.1

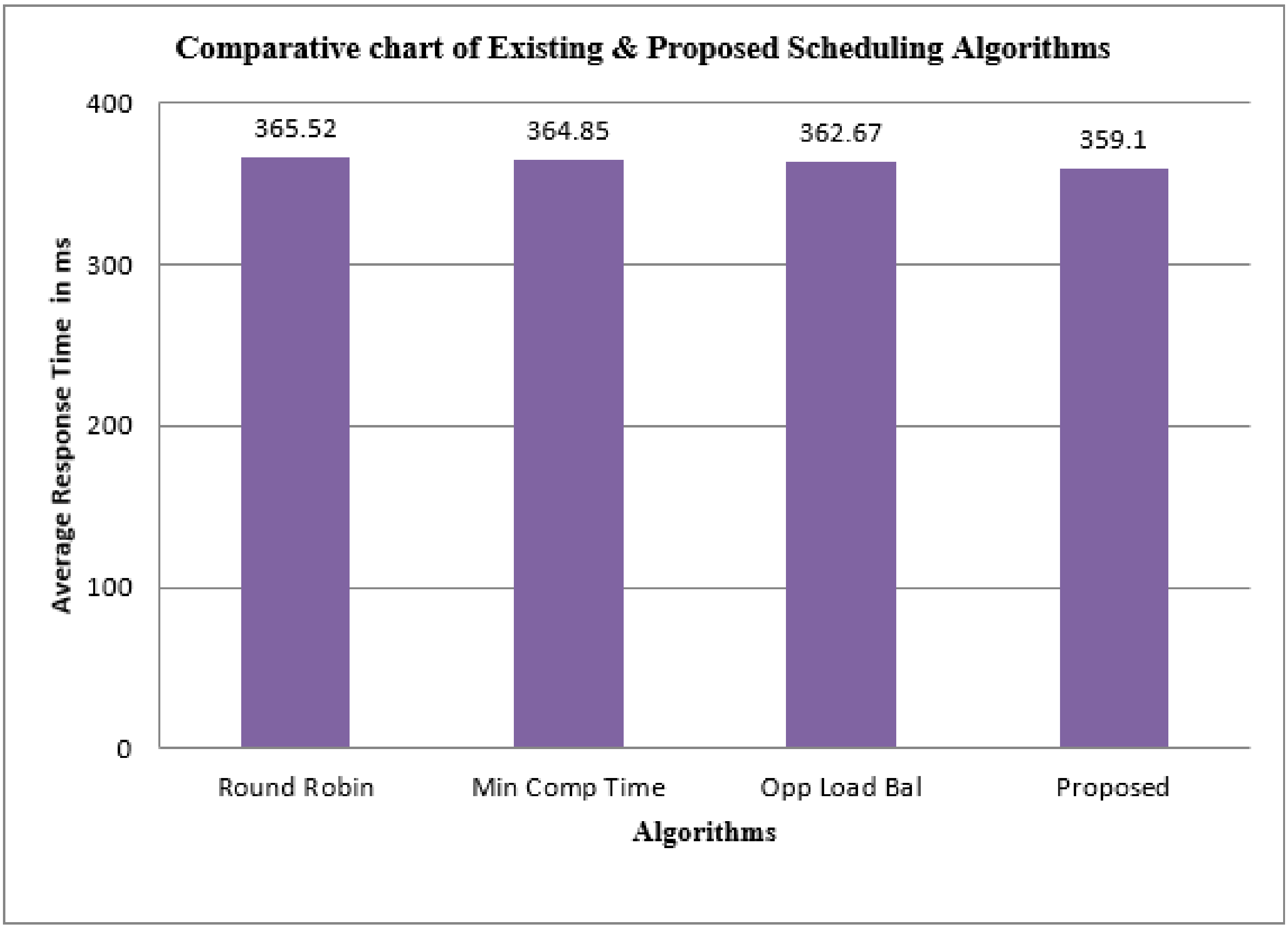

Figure 4 (Above): The comparison table of old techniques and our proposed scheduling algorithm for the average response time metrics.

Table 3 and Figure 5 below show the comparison table of old techniques and our proposed scheduling algorithm for the execution time metrics.

Table 3: The comparison table of execution time metrics.

No. of vms	Round Robin	Minimum Completion Time	Opportunistic Load Bal	Proposed Algo
5	0.5	0.4	0.35	0.3
10	0.7	0.6	0.55	0.5
15	0.9	0.8	0.75	0.7
20	1.1	1	0.9	0.8
25	1.3	1.2	1.1	1
30	1.5	1.4	1.3	1.2

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

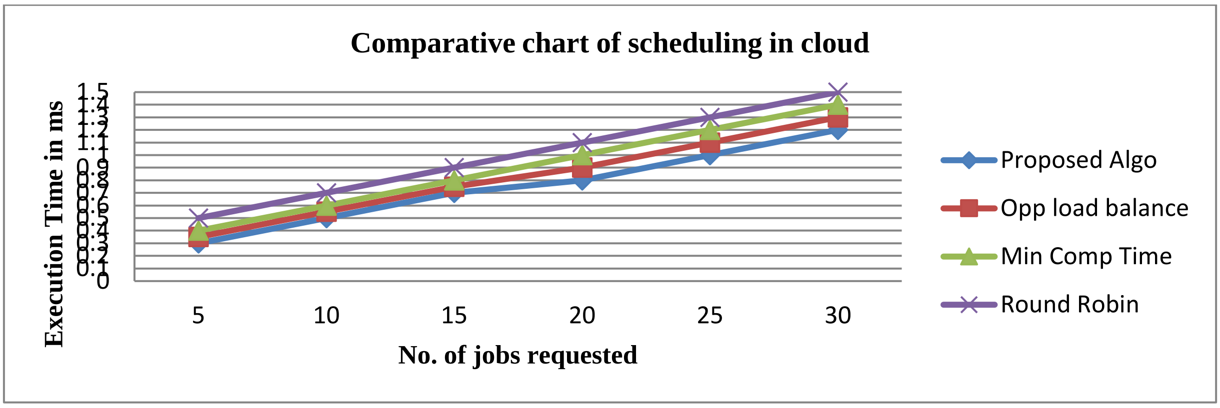

Figure 5: The comparison table of old techniques and our proposed scheduling algorithm for the execution time metrics.

As shown in above figures, we can see that the throughput decreases as long as quantity of jobs increases for all the algorithms of scheduling. Also the execution time rises with increase in number of jobs for all scheduling algorithms. From the figures, it can also be concluded that the proposed algorithm outplays the other scheduling algorithms. This is because the proposed algorithm uses approach of minimum completion time scheduling and load balance which not only assign the jobs to servers with minimum completion time but also performs the load balance for overloaded servers.

V.CONCLUSION:

The issue regarding energy consumption of cloud server during processing of their routine work is experimentally analyzed. Here our focus was to investigate and reduce energy consumption with enhanced scheduling approach based on multi-layer architecture. We proposed and implemented a multi-layer job scheduling approach and requested load balancer for managing multiple job queues along with resources utilization over cloud network. We evaluated our work with various performance parameters viz. energy, network (processor) utilization and response (processing) time etc. From the result analysis; we can make out that our algorithm which we proposed; outperforms the other scheduling algorithms. This is because our algorithm uses approach of minimum completion time scheduling which not only assigns the jobs to servers with minimum completion time but also performs the load balance for overloaded servers.

REFERENCES

1. Ning Liu, Ziqian Dong, et al., 2013, "Task scheduling and server provisioning for energy-efficient cloud-computing data centers" in 2013 IEEE 33rd International Conference on Distributed Computing Systems Workshops.
2. Qi Zhang and Raouf Boutaba, et al. 2014 "Dynamic Workload Management in Heterogeneous Cloud Computing Environments"
3. Gupta, P.K. and N. Rakesh, 2010. "Different job scheduling methodologies for web application and web server in a cloud computing environment", Proceedings of the 3rd International Conference on Emerging Trends in Engineering and Technology, Nov. 19-21, IEEE Xplore Press, Goa, pp: 569-572. DOI: 10.1109/ICETET.2010.24
4. Vinod Kumar Saroha, Dr. Sanjeev Rana **"**Review: Analysis, Design and working performance of scheduling algorithms in Cloud Network" Pg. 51-in IJETAE (International Journal of Emerging Technology and Advanced Engineering) Vol-07, Issue-06, June 2017 (ISSN 2250-2459 online)
5. Vinod Kr. Saroha, Sanjeev Rana; "Towards Energy Efficient Multilayer job scheduling approach over a cloud Network" IEEE Digital Explore; International Conference on Power Control, Signals and Instrumentation Engineering (ICPC& I-2017)
6. N. Tolia, Z. Wang, "Delivering energy proportionality with non energy proportional
7. Systems".In proc-hotpower, San Diego, Dec. 2008.
8. Suman Rani, Vinod Saroha, Sanjeev rana "Hybrid approach of round robin, Throttle and equally spaced technique for load balancing in cloud Enviornment" IJIACS, Vol-06, Issue-08, August-2017, ISSN-2347-8616.
9. Divya, Anil Arora, "Resource Allocation Scheme On Multiple Clouds" published in National Conference On Advances In Computing, Communication and Information (NCACCI-2015) page No.28
10. Orgerie and L. Leffevre. "When Clouds become Green: the Green Open Cloud Architecture", Parallel Computing, 19:228 - 237, 2010.
11. X. Tang and S. Chanson, "Optimizing static job scheduling in a network of heterogeneous computers", In Parallel Processing International Conference 2000 pages 373 - 382..
12. Swachil Patel, Upendra Bhoi, "Priority Based Job Scheduling Techniques In Cloud Computing: A Systematic Review", International Journal Of Scientific & Technology Research VOLUME 2, ISSUE 11, NOVEMBER 2013.
13. S.Kandula,and I.Stoica. "What slows datacenter jobs and what to do about it".RAD Lab, CA, January 2010.
14. Komal, Vinod Saroha, Sanjeev Rana, "An efficient service allocation and deallocation in cloud environment" IJRASET, Volume-05, Issue-08, July-2017.
15. Lipsa Tripathy, Rasmi Ranjan Patra, "Scheduling In Cloud Computing", International Journal on Cloud Computing: Services and Architecture (IJCCSA) ,Vol. 4, No. 5, October 2014.
16. Nima Jafari Navimipour and Farnaz Sharifi Milani, "Task Scheduling in the Cloud Computing Based on the Cuckoo Search Algorithm", *International Journal of Modeling and Optimization, Vol. 5, No. 1, February 2015.*
17. Surveillance System for monitoring in Cloud Computing using IOI", Nikita, Vinod Saroha, Sanjeev Rana, IJRASET, Vol-05, Issue07, July 2017.

AUTHORS PROFILE

Mr. Vinod Kumar Saroha is working at the post of Assistant Professor in Computer Science & Engineering and Information Technology department of Bhagat Phool Singh Mahila Vishwavidalaya; which is a state government women university and is located at Khanpur Kalan, Sonipat, Haryana. He obtained his Master of Computer Applications and Master of Technology from Maharishi Dayanand University, Rohtak. Presently he is pursuing his PhD in Computer Science and Engineering from M M Deemed University, Mullana-Ambala. He has more than 20 research papers in various International journals/conferences to his credit. His research interest includes Cloud Computing, Wireless Network, Network Security and Software Systems.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.

Dr. Sanjeev Kumar Rana is working as a professor, Computer Science & Engineering department at Maharishi Markandeshwar (Deemed To Be University), Mullana-Ambala, India. He obtained his Bachelor of Technology (Computer Engineering) from Kurukshetra University, Kurukshetra in 1999. He obtained Master of Technology (Information Technology) from University school of Information Technology at Guru Gobind Singh Inderprastha University, Delhi in 2007. He earned his PhD degree from M M University, Mullana-Ambala in 2012. He is also a CISCO certified instructor. He has more than 50 research papers in various International journals/conferences to his credit. His research interest includes Cloud Computing, Wireless Network, Network Security, MANETs, Sensor Network.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)
© Copyright: All rights reserved.