Energy Efficient Virtualized Scheduling and Load Balancing Algorithm in Cloud Data Centers.

J.K.Jeevitha[1], Dr.G.Athisha[2]

[1]*Department of Information Technology,PSNA College of Engineering and Technology.*

Dindigul,Tamil Nadu, India

jeevitha@psnacet.edu.in

[2]*Department of Electronics and Communication Engineering,PSNA College of Engineering and Technology.*

Dindigul,Tamilnadu, India

selvajeeva31@gmail.com

Abstract

Energy efficiency is one amongst the best most and important challenge in cloud Data centers. The datacenter is the highest resource place in cloud computing. The improper and unformed resource management leads to high energy consumption in cloud data centers. Scheduling and Load balancing techniques is doing the resource management in cloud forms. Bring about with better load balancing algorithm improves the excel in the resource assignment task. So as to scale back the energy consumption, this paper proposed three algorithms, the first one is identifying the load balancing factors and redistribute the load, the second one is find out the most suitable server to assigning the task to the server, achieved by Most Efficient First Fit Algorithm (MEFFA) and the third algorithm is processing the task in the server in an efficient way by Energy efficient Virtual Round Robin (EEVRR) Scheduling Algorithm with FAT Tree topology architecture. This EEVRR algorithm improves the Quality of Service via; steps send the task scheduling performance and cut the delay in Cloud data centers. Increase the energy efficiency by achieving the Quality of Service (QOS).

Key words: Cloud Computing, Data Centers, Task Scheduling, Load Balancing.

1. INTRODUCTION

Cloud Computing is one of the best research directions, it's the aggregate of utility computing, grid computing and distributed computing. The data processing and storages are highly increasing in the current computing world. So it guides the expressive growth of Data forms (Data Centers). Cloud Data centers offers various the types of services. It provides resource as a service that is an Information Technology (IT) Server, Networking components in the data center, Storage, Cooling system, etc., those hardware components are coming under the Infrastructure as a Service (IaaS). Software Components are such as social networks, search engine, software tools, Computing applications, etc are consider, as Software as a Service (SaaS) [8]. Cloud computing technology has resulted in maintaining large-scale Data centers consisted of thousands of computing nodes that consume ample amounts of electricity. Consistent with report of NRDC (Natural Resources Defense Council) the nationwide Data Centers used 91 billion KWH (Kilo Watt Hours) of energy consumption in 2013, and it's estimation to reach around 139 billion of kilowatt-hours by 2020, that is 53% to increase compared to today's consumption [http://www.nrdc.org/energy/datacenter- efficiency-assessment.asp].

In another report said that, data center servers power consumption is only between 10 % to 15 % of supplied electricity [http://www.datacenterjournal.com/it/industry-outlookdata- center-energy-efficiency/].One of the primary reasons for the high electricity consumption is incapable usage of the Data center resources. Due to the narrow dynamic power range of servers, it has seen that, even idle servers consume about 70 % of their peak power [11]. Idle servers consume more than 70% of energy consumption compared to active server. The proper server load balancing improves the resource utilization. Efficient resource usage increases the idle server count. And then ideal servers put into as switch off mode. The power off method of idle server increases the energy efficiency among the servers in the cloud data center. Other related factors to the energy consumption are delay and Quality of Service (QoS).

In order to scale down the energy consumption in cloud data center, this paper proposes three main tasks, the primary one is characteristic the load balancing factors and redistribute the load, the other is find out the most suitable server, to assigning the task to the server, achieved by Most Efficient First Fit Algorithm (MEFFA) and also the third algorithm is process the task within the server in an efficient way by Energy efficient Virtual Round Robin (EEVRR) Scheduling Algorithm. To boot this paper, focuses FAT Tree topology architecture in Data Centers.

The best task scheduling approach lower the energy consumption, improves the resource usage, shorten the make span time and it additionally cut down the delay among cloud servers. Scheduler has logic to seek out the most suitable server and assign the task to the server, based on the EEVRR algorithm. Another one task is load balancing that load balancer balance the load on the servers within the data centers. Task transferred from extremely used server to the lowly utilized server and a few of the underutilized server's that means that the tasks redirected to the excellent servers. Currently the idle servers are turned off. This paper compares the proposed algorithm with Round Robin (RR) and Weighted Round Robin (WRR) Algorithm. Experimental results show the proposed algorithm performance. The remaining part of the paper is, section 2 mentions the related work, in section 3 discuss proposed work, section 4 shows the mathematical model of the paper, section 5 shows the Experimental results and comparison, and section 6 discussed Conclusion and section 7 addresses the reference.

2. RELATED WORK

This article proposed Load balancing based on Bayes and clustering Algorithm (LB-BC) for cloud data center energy efficiency. In this paper combines Bayes and clustering Algorithm. This paper aims to satisfy long-term load balancing in physical host, via it achieved the network load balancing of cloud data center and place forwarded to efficient performance of external services. LB-BC can notice the appropriate physical host and effectively deploy the task to corresponding physical host [1].

This paper proposed Inter Cloud Manager (ICM) job dispatching algorithm for large-scale cloud. ICM had performed two tasks, such as Clustering and Decision making. For clustering purpose ICM used the hello packets, that collected the data from neighbors and decision-making function occurred for jobs dispatching, these techniques reduced the energy and power consumption among the cloud data center [2].

This paper suggested passive optical cross-connects networks (POXNs) for cost saving, power efficiency and reliable communication with the data center. POXNs supported in warehouse-scale data centers, avoided the addressing problem and improved physical-layer scalability challenges by using advanced interconnection techniques. [5].

This paper focused on three kinds of services like Cloud Content Delivery Service, Storage as a service (SaaS), and Virtual Machine placement in cloud data centers. Initial service, projected as Mixed Integer Linear Programming (MILP) model to progress Cloud Content delivery service. Second service developed an energy efficient cloud content delivery heuristic, DEER-CD, with a comparable power efficiency to the MILP results extend Content Delivery model to optimize the Staas, Third service, optimize the virtual machine placement ,to cut the 25% of power consumption, done by Virtual machine(VM) Slicing methodology[6]

This paper proposed server consolidation technique within the cloud data centers. The author divides the work into two modules:(1) consolidation planning module that calculated a set of workloads, minimizes the number of Physical Machines for task assignment by an number programming model, and (2) migration planning module that, targeted source Virtual Machine(VM) assignment scenario and target VM assignment scenario, reduced the number of VM migrations by a polynomial time algorithm, each modules bonded the performance losses of assorted workloads below configurable thresholds.

Reduction of the number of physical machines led to lower power consumption the low power consumption led to higher energy potency [7].

This paper introduced a model of task scheduling for analyzing data center energy efficiency in cloud computing. Task assignment to the server carried out by integer programming problem, for reducing the energy consumption in data centers. Here the author mentioned most efficient Server first (MESF) scheduling algorithm. Here, this algorithm appointed the task to the foremost energy efficient servers. The centralized scheduler sorted the servers supported the energy efficiency and keep within the list. For a data center with one server type, MESF appointed the number of tasks to every efficient server, till the saturation point (where the server performance decays considerably, or the task queuing delay is approaching its delay constraint) was reached. The central scheduler maintained a sorted list of unsaturated and active servers with their energy profiles. The servers sorted in keeping with their energy profiles wherever the foremost energy-efficient servers placed on the top of the list. Upon received task requests, the scheduler allotted tasks to the servers from the sorted list from top to bottom. The servers received task assignments information and their energy profile updated information. Once the foremost energy-efficient servers saturated, they removed from the list till they become unsaturated [8].

In this paper, the author proposed Virtual Machine (VM) Migration technique whereas the host acquired overloaded or under loaded, used one of the algorithm was referred to as PABFD (Power Aware Best work Decreasing) algorithm for putting the VM to best host. This algorithm targeted two varieties of parameters the first one, CPU utilization and another one, power consumption of physical machine. Based on the two parameters, it allotted the VM to host so as to reduce the power consumption. This method managed to induce lower power consumption, less quantity of SLA violation and less amount of performance degradation [9].

In this paper proposed inter Cloud Job (ICM) dispatching algorithm for large scale cloud surroundings. ICM principally performs two tasks: (1) clustering and (2) decision making. “Hello” packets used to collect system load and therefore the end-to-end delay from the client to the host. A hello packet was, sent periodically on every network interface to discover and checked the connections among neighbors. Hello packets were broadcasting, to enable a dynamic router and host server discovery. The ACK packet was sent by the receiving server (destination) and left back to the sending server (source). Each interconnected server’s information’s were periodically updated by hello packet through ACK packet that was, being sent by the destination server. Once the supply received the ACK packet from the destination, it computed and hold on the network link (hop count, delay, loss) and system load (job execution time, memory usage, range of waiting jobs) information. Every cloud host might create its own “cluster” based on the information. Better load balancing and decision making techniques improved QoS and energy efficiency [10].

In this paper articulated the Round Robin (RR) scheduling algorithm was designed to carry the distribution of CPU time among the scheduled tasks. On an identical context, all the tasks acquired on a queue list, whereas every task obtained a small unit of CPU time [13].

In this paper, introduced a new perception to simple Round Robin (RR), that compared the remaining burst time of a task to the algorithmic program static time quantum once the primary allocation. If the remaining time was smaller than just once quantum the CPU could apportion the time quantum to the task otherwise it's sent to a waiting queue [14].

In this paper considered Weighted Round Robin (WRR) scheduling for up the energy potency in cloud information centers with Dynamic voltage frequency scaling (DVFS). Energy efficiency was calculated by the CPU utilization of servers [15].

In above cited techniques were concentrated either only the scheduling or either load balancing. Therefore this paper combined both scheduling and load balancing schemes. In above stated scheduling algorithm RR and WRR concentrated only the least time consuming process, here the proposed EEVRR algorithm preferred the priority to the small amount of time quantum consuming process and this tied up with the FAT Tree topology architecture.

3. PROPOSED WORK

3.1 Data center topology architecture network

In cloud computing Data Center Network (DCN) is considered as a vital component.DCN designed with large number of hosted servers and switches with high performance and communication link. The FAT Tree topology architecture used commodity switches in cloud data forms, that can support any communication pattern with full bisection bandwidth [8][17].

3.2 Load Balancing and scheduling

Here, the load balancing was carried out in the cloud data center level. Load sharing or Load balancing technique used to segregate the amount of work that work had done between two or more servers. So that more works completed at the same time and in general, all users served faster. In Fig.1, Shows the architecture of the proposed work. at the start the clients sent the user request from any kind of interface to the cloud server's forms. In between the requests received by the scheduler for correct assignment of user request to the right and stable server. The scheduler did the very important job within the architecture. Scheduler assigned the user request to the optimal server depends upon the load capacity of the servers. The scheduler receives the load information from the info gatherer module. It was maintained by the network load balancer. Info gatherer used to collect the data sent by user request, and it also had the load information about the servers in the data center.

The network load balancer module contained two sub modules, one was network manager and another one was load balancer. The network manager, responsible for setting the network configuration and it conjointly maintained the server clustering. Sever clustering formed based on the server load capacity. Load balancer identified highly utilizing and underutilized servers. Highly utilized server known as saturated server and underutilized server known as unsaturated server. Redistributed the load from highly utilized to underutilized servers, and then the load balancer found if any other very poorly utilized node left out.The very poorly utilized server's load transferred to any one of the unsaturated servers. Next, the load balancer switched off the idle servers, because the idle server consumes more energy compared to active server. This method enhances the proper resource utilization.

3.1.1 Identifying Load balancing factor and redistributing the load

Sum of the server load in a data center a state as

$${}_{j=1} \qquad (1)$$

j denoted as number of servers in datacenters.
Calculating load capacity of the server:
Load per unit capacity:

$$LPC = \frac{L_o}{\sum_{i=}^{m} c_j} \qquad (2)$$

Threshold Ti $= LPC * C_j$ (3)

C_i represented as the capacity of the server

Load balancing factors were calculated by the threshold value Ti. Algorithm 1 replicated the load determination and load sharing depends on the load factors of the various servers. Server load compared to the threshold value. First Condition, if the server load, above the value of Ti, redistributes the task from heavily loaded server to lightly loaded server.

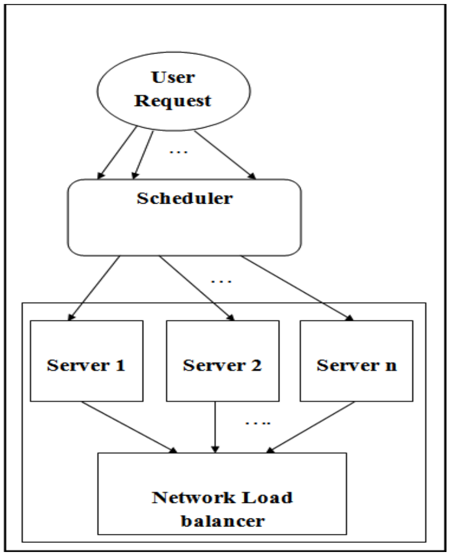

Fig.1 Architecture of Scheduling and Load balancing

The second condition held the two cases. Case 1, if the load, below the threshold value, then server considered as under loaded. Case 2 servers utilize only the 0.2 percent of server capacity that means the server very poorly utilized the CPU resources, so transfer the task from poorly utilized server to any one of the lightly utilized server and then switch off the idle server. The third condition, if the server load is equal to the Ti value, it is balanced.

Algorithm 1 Load balancing Algorithm (LBA).

```
Compute Server load $L_s$ and Capacity of the server $C_s$ .
Find the Threshold value $T_i$ .
If( $L_s > T_i$ )
Serv $\leftarrow$ over loaded,
Transfer the task from heavily loaded server to lightly loaded server.
Else if ( $L_s < T_i$ )
 If( $L_s$ < 0.2% 0f $T_i$ )
Server $\leftarrow$ very poorly utilized.
This server task is transferred to any one of the under loaded server.
Switch off the server.
End if
Server $\leftarrow$ under loaded.
Else
Server is balanced.
End if
```

3.2Task assignment and task Processing

Network manager stored all servers in the linked list based on the capacity of the server in sliding order. The server list was altered automatically based on the capacity of the servers, very poorly utilized server took away from the list, after that it put into the power off mode. While the user request were increasing, all listed servers were very active or properly utilized, when the scheduler needed additional servers, power off servers obtained back into the power on mode. This server included in the list. The server list information sent to the scheduler via info gatherer. In the fig. 2 explains the Energy Efficient Virtual Round Robin (EEVRR) algorithm. The process (request) entered into the scheduler queue in First Come First Server (FCFS) basis. Scheduler queue held the every incoming process. Scheduler queue created as a circular buffer. And the scheduler stored the incoming task in sliding order based on the highest resource would like.

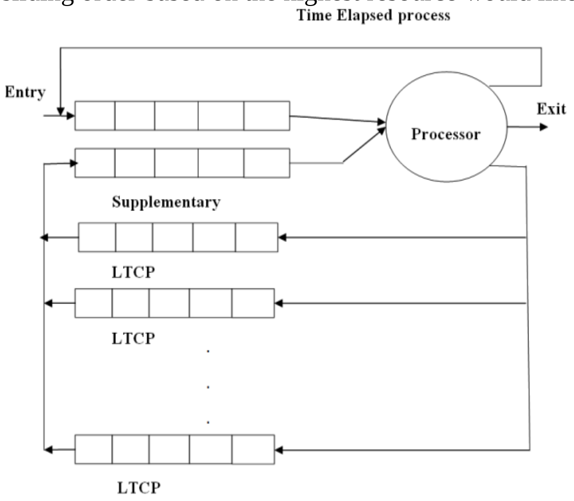

Fig.2 EEVRR Scheduling

After that, scheduler mapped the task to the server. Algorithm 2 Most efficient first fit algorithm had done the task assignment. The EEVRR algorithm allocated the time quantum or time slice to the every process. The process should be completed, their task within the particular time slice. If the process execution could not be completed within the time slice, the process will be re-entered into the scheduler queue in front. Suppose if the process did not complete their IO process or least amount time consuming process means, the process put in the least time consuming process queue, after that it would be switched to the supplementary queue.

EEVRR algorithm gave higher priority to supplementary queue process. EEVRR algorithm compared with a Round Robin (RR) algorithm and Weighted Round Robin (WRR) algorithm. EEVRR overcome the downside of RR algorithm. The RR algorithm failed to assign the priority of the IO bounded process. So the RR algorithm ended up with heavy starvation. Scheduler finished their jobs with EEVRR algorithm. Algorithm 3 portrayed the Scheduling process.

Algorithm 2 MEFFA

For

```
ser ← list the server
$T_s$ ← list the task
Do
Assign the $T_s$ to Serv
Assign the task to server in the first fit manner
While (End of the task)
End for
```

Algorithm 3 EEVRR

```
Begin
N number of Process P arrives
P enter in to the ready queue
Allocate the Time Slice to each task
P  loaded in to the server from ready queue
If (P executed within the time slice)
Exit the process from the ready queue.
Else if(P is CPU bounded Process)
P  put into the ready queue.
Else if (P is LTCP)
P  put into the LTCP queue.
Next P Moves to supplementary queue
Endif
End
```

4. MODEL

Let S = (S1,S2, . . . ,Sm) be the set "m" server, which should process "n" number of tasks represented by the set T = (T 1, T2, . . . , Tn). All the servers are running in parallel, In that modeling, n tasks assigned to m servers, are produced by the linear programming model. Here the processing time is considered as PRT_{ij} task i is assigned to server j, x_{ij} is number of task i is assigned to server j.

The capacity of the server is, calculated by the number of processing elements in the server and million instructions per seconds of processing element, Here C_s is indicated as Capacity of the server. Pe_{num} is number of processing elements and Pe_{mips} is millions of instructions per seconds.

$$C_{s=}Pe_{num} * Pe_{mips} \qquad (4)$$

The capacity of the overall servers in the data center is calculated by the following ,

$$C = \sum_{j=1}^{m} C_{sj} \qquad (5)$$

Utilization of the overall server is U that is utilization is enumerated based on requisite of the server and the capacity of the server. Requisite is assessed by the busy time of the server (consider CPU) and Capacity can be assessed by one another way, that is inclusive of busy time and idle time of the server.

$$U = \frac{\sum_{j=1}^{m} R_{sj}}{\sum_{j=1}^{m} C_{sj}} \qquad (6)$$

R_{sj} is the Requisite of the server and C_{sj} is the Capacity of the server.

$$Po = K.P_{max} + (1-K).P_{max}.u \qquad (7)$$

$$K = {P_{idle}}/{P_{max}} \qquad (8)$$

Po is considered as Power consumption of the server in the data center. P_{max} is power consumption at the maximum performance and P_{idle} is the power consumption at the idle time period of the server.

$$E = \sum_{i=1}^{n}\sum_{j=1}^{m} PRT_{ij}\, x_{ij} Po_{ij} \qquad (9)$$

By using the linear programming model, calculate the Energy (E) of the servers in data center by using the parameters of processing time, number of tasks and power consumption of the server.

5. EXPERIMENTAL RESULTS

In this section, we have a tendency to given the performance analysis of planned EEVRR task scheduling scheme with exiting RR and WRR scheduling algorithm. we tend to modeled a FAT Tree data center with EEVRR scheduler, variety of task and variety of servers in Cloud Sim simulation with java primarily based solution to guage the energy consumption. Data center designed with number of inter connected server in cloud sim simulator. During this experiment totally designed 5 data centers each data center consists of twenty five servers with in numerous configurations and designed incoming jobs twenty five to fifty tasks request. EEVRR algorithms evaluated with the subsequent parameters. 1. response time, 2.Waiting Time, 3.Task Completion Time, 4.Delay, and 5.Energy Consumption. All the parameters compared with RR, WRR and EEVRR among MEFFA in addition as load identification and load balancing factors by using various heterogeneous server nodes. In fig.3, the x-axis shows the number of tasks inward, and therefore the y-axis shows response time in seconds. Response time is that the measure between the task submission and task had taken within the process. Here EEVRR response time enlarged compared to different two algorithms. Initially, all algorithms are at identical level, once the quantity of tasks enlarged as twenty, the reaction time of the RR and WRR attain the minimum value. Within the fig.5, x-axis mentions the number of task and y-axis mention the waiting time in seconds. The EEVRR algorithm's waiting time was very smallest, compare to other two algorithms. As a result of EEVRR formula provides higher priority to LTCP. The RR and WRR don't consider the LTCP process. In the Fig.5 show the delay comparison. The EEVRR algorithm achieves low delay, compare to others. The EEVRR algorithm selects the optimal servers throughout the task assignment, thus it will meet the low delay.

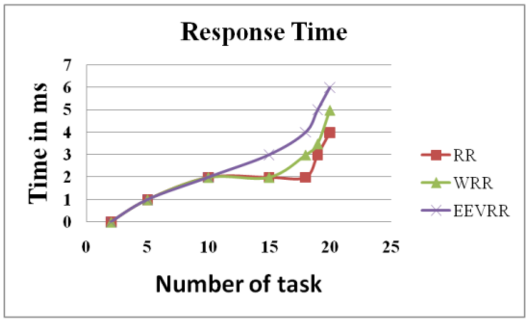

Fig.3 Comparison of Response Time

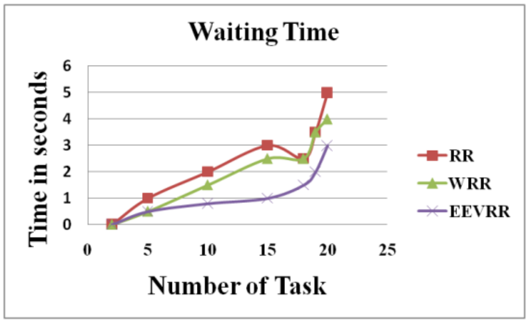

Fig.4 Comparison of waiting Time

The Fig.6 shows the energy consumption in the servers. In the X-axis denotes number of servers, In Y-axis denotes energy consumption in units. The units of power can be converted into watts. One unit equals to 10 watts. Here the energy efficiency compared with three scheduling algorithms. The EEVRR algorithm Consumes less amount of energy. This projected energy, calculated with the help of a number of tasks, processing time and power of server during execution of tasks. Already this scheme reduces the delay, waiting time and increase response time. So it automatically leads to low energy and power consumption.

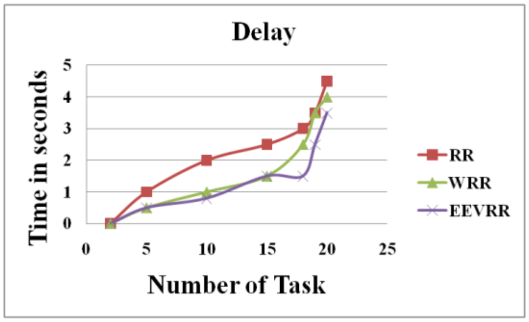

Fig.5 Delay Comparison

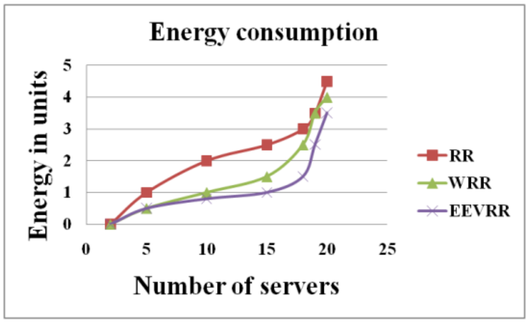

Fig. 6 Energy efficiency comparison

6. CONCLUSION

This project introduced three algorithms for energy efficiency, such as LBA, MEFFA and EEVRR with FAT Tree topology architecture. The framework designed based on the load balancing and scheduling in the cloud data center. The first one is load identification and load distribution, second one is MEFFA used for selecting best efficient server, third one is EEVRR scheduling algorithm, and it did most efficient jobs for energy reduction. The LBA algorithm mainly used for load redistribution, based on the threshold value and it classified load according the three factors. MEFA algorithm used to select the most suitable servers, and the EEVRR algorithm utilized for scheduling and processing. The EEVRR algorithm played a major role in this project, because it gives higher priority to LTCP project. This algorithm compared to RR and WRR Algorithm. In this paper graph show the Comparison of The EEVRR scheduling algorithm with two algorithms, Experimental results shows performance of three algorithms. The EEVRR algorithm obtained best results are compared to remaining two. In the future work, during the experimental analysis, the number of tasks will be increased and consider number of data centers with significant severs will be going to apply in the framework and EEVRR algorithms compared with another specific set of algorithms.

7.REFERENCES

1. Jia Zhao, Kun Yang, Xiaohui Wei, Yan Ding, Liang Hu, and Gaochao Xu, “A Heuristic Clustering-Based Task Deployment Approach for Load Balancing Using Bayes Theorem in Cloud Environment”, IEEE Transactions On Parallel And Distributed Systems, Vol. 27, No. 2, February 2016.
2. Byungseok Kang*and Hyunseung Choo,” A cluster-based decentralized job
dispatching for the large-scale cloud”, EURASIP Journal on Wireless Communications and Networking,2016:25.
3. Anton Beloglazov and Rajkumar Buyya,” Managing Overloaded Hosts for DynamicConsolidation of Virtual Machines in Cloud Data Centers under Quality of Service Constraints”, IEEE Transactions On Parallel And Distributed Systems, Vol. 24, No. 7, July 2013.
4. Hui Zhang, Guofei Jiang, Kenji Yoshihira, and Haifeng Chen’” Proactive Workload Management in Hybrid Cloud Computing”, IEEE Transactions on Network and Service Management, Vol. 11, No. 1, March 2014.
5. Wenda Ni, *Member, IEEE*, Changcheng Huang, *Senior Member, IEEE*, Yunqu Leon Liu, Weiwei Li, Kin-Wai Leong, and Jing Wu*, Senior Member, IEEE,”* POXN: A New Passive Optical Cross-Connection Network for Low-Cost Power-Efficient Datacenters*”,* Journal Of Lightwave Technology, Vol. 32, No. 8, April 15, 2014.
6. Ahmed Q. Lawey, Taisir E. H. El-Gorashi, and “Jaafar M. H. Elmirghani” Distributed Energy Efficient Clouds Over Core Networks”,Journal Of Lightwave Technology, Vol. 32, No. 7, April 1, 2014.
7. Kejiang Ye, Zhaohui Wu, Chen Wang, Bing Bing Zhou, Weisheng Si, Xiaohong Jiang, and Albert Y. Zomaya, ” Profiling-Based Workload Consolidation and Migration in Virtualized Data Centers”, IEEE Transactions On Parallel And Distributed Systems, Vol. 26, No. 3, March 2015.
8. Ziqian Dong, Ning Liu and Roberto Rojas-Cessa,” Greedy scheduling of tasks with time constraints for energy-efficient cloud-computing data centers” Journal of Cloud Computing: Advances, Systems and Applications (2015) 4:5
9. Mohammed Rashid Chowdhury, Mohammad Raihan Mahmud and Rashedur M. Rahman,” Implementation and performance analysis of various VM placement strategies in CloudSim”, Journal of Cloud Computing: Advances, Systems and Applications (2015).
10. Byungseok Kang and Hyunseung Choo,” A cluster-based decentralized job dispatching for the large-scale cloud”, EURASIP Journal onWireless Communications and Networking (2016) 2016:25
11. Fan X, Weber WD, Barroso LA (2007) Power provisioning for a warehousesized computer. Proceedings of the 34th Annual International Symposium on Computer Architecture (ISCA 2007), ACM New York, NY, USA, pp 13–23.
12. D. Chitra Devi and V. Rhymend Uthariaraj” Load Balancing in Cloud Computing Environment Using Improved Weighted Round Robin Algorithm for Nonpreemptive Dependent Tasks”, Hindawi Publishing Corporation The Scientific World Journal Volume 2016, Article ID 3896065.
13. G. T. Hicham, EL.Chaker, Cloud Computing CPU Allocation and Scheduling Algorithms using CloudSim Simulator, *International Journal of Electrical and Computer Engineering*, Vol 6, No 4, 2016
14. M. Manish Kumar, K. Abdul Kadir, AN IMPROVED ROUND ROBIN CPU SCHEDULING ALGORITHM, *Journal of Global Research in Computer Science*, Volume 3, No. 6, June 2012
15. Abdulaziz Alnowiser Eman Aldhahri Abdulrahman Alahmadi Michelle M. Zhu “Enhanced Weighted Round Robin (EWRR) with DVFS Technology in Cloud” Energy-Aware Computational Science and Computational Intelligence (CSCI), 2014 International Conference
16. R.Leena Sri1, Dr.N.Balaji “ Speculation based Decision Support System for Efficient Resource Provisioning in Cloud Data Center” International Journal

of Computational Intelligence Systems, Vol. 10 (2017) 363–374”

17. Lin Wang, Fa Zhang, Jordi Arjona Aroca, Athanasios V. Vasilakos” GreenDCN: a General Framework for Achieving Energy Efficiency in Data Center Networks” IEEE journal on selected areas in communications, vol., no. , January 2014