An Approach based Upon Cross Breed Algorithm To Improve Job Scheduling at Cloud

[1] **Varinder Kaur**, [2] **Gurjot Kaur**

[1, 2] Computer Science Engineering, Chandigarh University, Gharuan, 160055, India

Abstract - In the cloud computing, many of the users send requests to cloud at the same time to access services. Thus, a big challenge of scheduling of these tasks is in cloud computing. Many algorithms like FCFS, SJF, Priority based, RR, MLQ, LSTR used to schedule the tasks in cloud computing. In cloud computing, most of the data centers consume vast amount of energy and take much more time to schedule the jobs. In this research paper, deploy a hybrid algorithm for job scheduling in cloud computing, using the combination of Multi-Level Feedback Queue Scheduling and Least Slack Time Rate (LSTR) is proposed to improve the issue of maximum energy consumption and time consumption by the data centers. Least Slack Time Rate is used to first select those processes that have the smallest "slack time". Multi-Level Feedback Queue is used in this scheme the processes can move between the different queues. MLFQ uses the working principle of Round Robin and First come First Serve scheduling algorithms. The performance of the proposed method is measured by calculating the parameters of Energy Consumption, Minimize Processing Time, Executed Jobs, and Unexecuted Jobs.

Keywords - *Cloud Computing, Job Scheduling, Least Slack Time, And Multi-Level Queue And Priority Algorithms.*

1. Introduction

Cloud Computing is large scale distributed computing paradigm. Cloud computing afford the application which is delivered as services over the internet and the hardware and software system in the data centers that provide those services [5]. Now days, most of business organizations and educational institute use the cloud environment. When we open an IT magazine or open any websites then cloud is anywhere. Amazon play a key role in cloud computing.

Some characteristics of cloud include [7]:

- On-demand self-services
- Ubiquitous network access
- Location independent resource pooling
- Rapid elasticity
- Pay per use

In the cloud computing, many of users send requests to cloud at same time for access services. So, a big challenge is to scheduling of these tasks on cloud computing. The main goal of scheduling is to minimize the various parameters like CPU utilization, turnaround time, waiting time and throughput [7]. Scheduling of jobs is a mapping mechanism from users' tasks to be appropriate selection of resources and its execution [7]. Job scheduling is flexible in nature.

Need of job scheduling in cloud computing is discussed as follows:

- **Load Balance–** In the cloud environment load balancing and task scheduling has very closely related to each other. The algorithm of job scheduling can maintain the load balancing. So load balancing is an important measure in job scheduling.

- **Throughput of the system –** To conclude the performance of system in job scheduling measure throughput. For both users and cloud providers increase throughput is benefit.

- **The best running time –** in cloud, according to needs of users jobs are divided into various categories, and then on the basis of different goals set the best running time for each job. In a cloud environment task scheduling indirectly improve the QOS (Quality of Service).

- **QOS (Quality of Service) -** in cloud computing, when the jobs of the job scheduling comes to job allocation. It is compulsory to guarantees about QOS of resources [5].

2. Literature Survey

Lizhe Wang, et al. [34] gave an idea to categorization of green computing conformance measurement in data centers, such as power metrics, thermal metrics and extended performance metrics. According to the paper two methods are used to make a green data centers. That is Green, Greenfly. It also identified carbon emission per unit time. Andrew J. Younger et al. [24] described a frame work for efficient green enhancement in cloud architecture. It is based on power aware scheduling, variable management and minimal virtual machine design. It has improved overall system efficiency. It is used to evaluate the performance and overall capacity of virtual machine by using power based scheduling of virtual machine. Antow Beloglazov et al. [3] have explained efficient resource utilization. They have proposed best fit decreasing that is modification best fit algorithm. This method is more efficient, but it is complex as it provides quality of service by dynamic reallocation of virtual machines.

Wang et al. [18] explained the method to predict a workload on a data center. The researcher used artificial natural networks that is, a machine learning technique. To perform experiment, data is collected from data centers. It decreases the complexity of data center by using thermal impact matrix. ANN based prediction technique reduce power consumption by using thermal aware workload scheduling algorithm. Yuetsu Kodama et al. [27] clarified a relationship among CPU load and fan speed.This paper also showed that the power which are consumed by fans could be reduced, when the rearrangement or scheduling of node are done**.** a new metric for data center power efficiency to fairly evaluate the contribution of each improvement for power efficiency. Delavan et al. [6] proposed a hybrid heuristic method to find scheduling for workflow graph. So to obtain quick response genetic algorithm is used. It also described load balancing of resources. Round Robin and Best Fit methods are used to create initial population.

Zhou Lei et al. [32] has explained an energy efficient scheduling approach. It is based on private cloud. To achieve multiple objectives, it is very difficult to schedule virtual machine. Virtual machine has many applications like virtual desktops and virtual libraries. This paper proposed an energy efficient scheduling algorithm based on least load first algorithm. It balances the load when data center is running on low power mode. But the problem is to powering down the busy nodes due to some threshold value is not feasible. Chihyun Jung et al. [12] This paper proposed a way to apply scheduling methods based on MILP models to problems which are too complex to solve in short times. The proposed decomposition method works very effectively with the suggested MILP for the performance. Amit et al. [15] explained the method to reduced execution time by using generalized priority scheduling algorithm and compare with FCFS and Round robin. To obtain quick response, cloud sim toolkit is used. IAAS provide service in the form of VMs. In this paper, job scheduling is used to control the order of work performed by the computer. Its main advantage is high performance computing and best system throughput. In this paper, VM scheduling is used for dispatching jobs. The main disadvantage of FCFS is that its response time and turnaround time is slow. Round robin is used for internal scheduling. Its main drawback is that largest job takes enough time.

Sudha Sadhasivam et al. [25] proposed an algorithm of mapping of tasks efficiently to available resources in cloud. This algorithm measures both cost of resources and computation performance. The problem with traditional task scheduling is to avoid the relationship between the various application bases. Tasks are grouped and sorted according to their calculated priority levels. They are placed in three different priorities lists namely, high, low and medium priority. Zhang Kai et al. [23] proposed an algorithm on time-cost based model. The time cost trust is based on subset tree algorithm. The subset tree algorithm is based on the principal of economics. This method mainly concerns with reliability factor. Only one factor from time, cost and trust cannot meet the user's demand. So, the time-cost model has been introduced to improve users demand.

Dipti Bhansali et al. [13] explained the mechanism of an Optimistic Differentiated Job Scheduling System. This algorithm is developed to serve the multiple requests. This method is proposed to handle multiple requests of services like uploading and downloading. Multiple requests are processed by use of non-primitive algorithm. Its main goal is to provide optimistic value of service. The users get the quality of service and service providers gain maximum profit. It exploits the under-utilized resources at non-peak times. Utilization of resources is done in transient way. This paper implements static load balancing based on size of files. Isam Azawi Mohialdeen [30] surveyed about the scheduling algorithm used in cloud computing. Scheduling is an important aspect to schedule the jobs on virtual machines. In cloud, single scheduling algorithm is not sufficient because single algorithm does not consider all performance metrics and Maintain quality of service. Many scheduling algorithm have been proposed to enhance the systems performance in terms of throughput and cost. This paper compares 4 types of scheduling algorithms namely RR, Minimum Completion Time, Opportunistic Load Balancing and Random Resource Selection**..** Each scheduling algorithm have performed superior on some metrics. All measuring characteristics are not provided by each and every algorithm. The selection of good scheduling is based on characteristics that fulfill the needs of customers as well as service provider. Huankai Chen et al. [21] studied about the genetic algorithm it was developed to scheduled task on virtual machines in efficient manner. This algorithm was made by the combination of Min-Min and Max-Min in Genetic algorithm. This technique was adopted to schedule task in order to reduce the execution time. This mechanism has achieved better performance than standard Genetic algorithm.

This survey [38] explained about the various algorithms to obtain optimal solution in job scheduling. The job scheduling was evaluated by cost and execution time. The various optimization techniques in this paper were based on these evaluations. These techniques are ant colony optimization and modified bee's life algorithm. Kun Li et al. [16] described a job scheduling policy in cloud base on load balancing ant colony optimization (LBACO) algorithm. The major goal of the entire system is to balance load while trying to reduce the performance of given task set. Ant colony optimization is an efficient dynamic task scheduling in a cloud. ACO is random search algorithm. LBACO outperformed FCFS and basic ACO. It is achieved good system load balance in any situation in any specific moment and take less time to execute.

Wang et al. [20] proposed an improved load balance is introduced to reduce the makespan and increase the resource utilization. It also improves Min-Min. But at same time cloud providers provide service as per demand of different users. Services were provided in terms of different levels of quality of services. This paper focused on efficient resource utilization, total completion time of task and user priority in a cloud.

Monica Gahlawat et al. [39] analyzing and evaluating the performance of various CPU scheduling in cloud environment using Cloud Sim, the basic algorithm of operating system like FCFS, Priority Scheduling and Shortest Job First test under different parameters that which scheduling policy perform better. R. Suchitra et al. [43] proposed a modified been packing algorithm for server con0solidation that avoids unnecessary migrations and minimizes the instantiation of new physical servers. They implement ideas from the First Fit algorithm for live migration of virtual machines. They have simulated our algorithm using java with multiple test cases. The proposed system consequently results in server consolidation through minimal migration.

N. Gupta et al. [40] discussed the work of various cloud providers and research groups that are working ahead in adding the advantages of cloud services. In spite of all the progress and technology enhancements that cloud computing brings, enterprises still face some challenging problems while reaching to a decision of whether to or not to adopt cloud.

3. Problem Formulation

In the cloud computing, many of users send requests to cloud at same time to access services. So to calculate these jobs many algorithms are used. But from literature survey conclude that these algorithms consume more energy and take more time for the scheduling of jobs

Most data centers, consume vast amounts of energy in an incongruously wasteful manner, interviews and documents show. As a result, data centers usually waste a lot of the electricity they pull off the grid, there more power consumption and wastage of time. So there is a strong need to optimize above three factors energy consumed, response time and no. of jobs executed per time.

The current research concern is to counter the unwanted energy and time consumed in data centers. This is gaining exceptional attention of researchers with respect to scheduling of the computing resources. In reality, Service providers make use of high quality use of IaaS and PaaS for developing their services without consideration of physical hardware. Thus, we can say that one of major issues in datacenters is to manage optimum energy, power usage in the systems.

In this research proposal, we will deploy hybridization of least slack time scheduling and multilevel feedback queue scheduling to achieve above problem.

4. Proposed Solution

From literature survey we have studied the various algorithms which are used for job scheduling. These algorithms have the problem with energy consumption and time. We proposed an algorithm which consume less energy and take less time for execution of tasks.

Table 1.1 Proposed Algorithm

```
• START K=0;
• Initialize Total Jobs=0;
• Initialize Job Selection = True;
• For Every Selected job Initialize
  Server Configuration = True
• Apply Job. Sort (Set. Sorting.
  Priority = True)
• For k=1: Selected job
• If System. Configuration. Meets.
  Job. Specification
• Job .Allocated = True; k=k+1;
• If Job. Completed == True
• Evaluate parameter values
• End
```

5. Work Done

Step 1: Start
It is the first step of execution that initializes the process of execution and loads all files and supported libraries in memory. After configure system with client it transfer control to next step that is either generate system or generate task.

Step 2: Generate System
After initialize all the content and libraries, system ready to perform their working. It requires nodes to generate network for execution.

Step 3: Generate Task
It is used to generate tasks for execution and get results from generated network of nodes with the help of proposed scheduling algorithm that schedule and generate results with different parameters.

Step 4: Properties
For generation of tasks and systems some properties are used to define their behavior during execution. These properties are like Ram, Processor etc. These are used to make a node behave as a system in network for execution task as real environment.

Step 5: Pre-Allocate task
This step defines the number of tasks that arrive in the process of scheduling. It manages them and makes a queue and allocates them.

Step 6: Limit
Now some time nodes are busy or their limit to execute task is over and new task occur for execution than proposed system get all the possible routes and their extended limits and broadcast them to network for refresh their working capacity and optimize their working.

Step 7: List Value
Now list Value is used to check that the entire step that perform before execution are successfully executes and system ready to allocate task on networks of nodes that are attached with scheduler.

Step 8: Broadcast
Here if some time system limit exceeds or a task is not executed. In this case system setup a virtual environment for creates a VM for their execution.

Step 9: VM
Here system allocates jobs to the VM and calculate results.

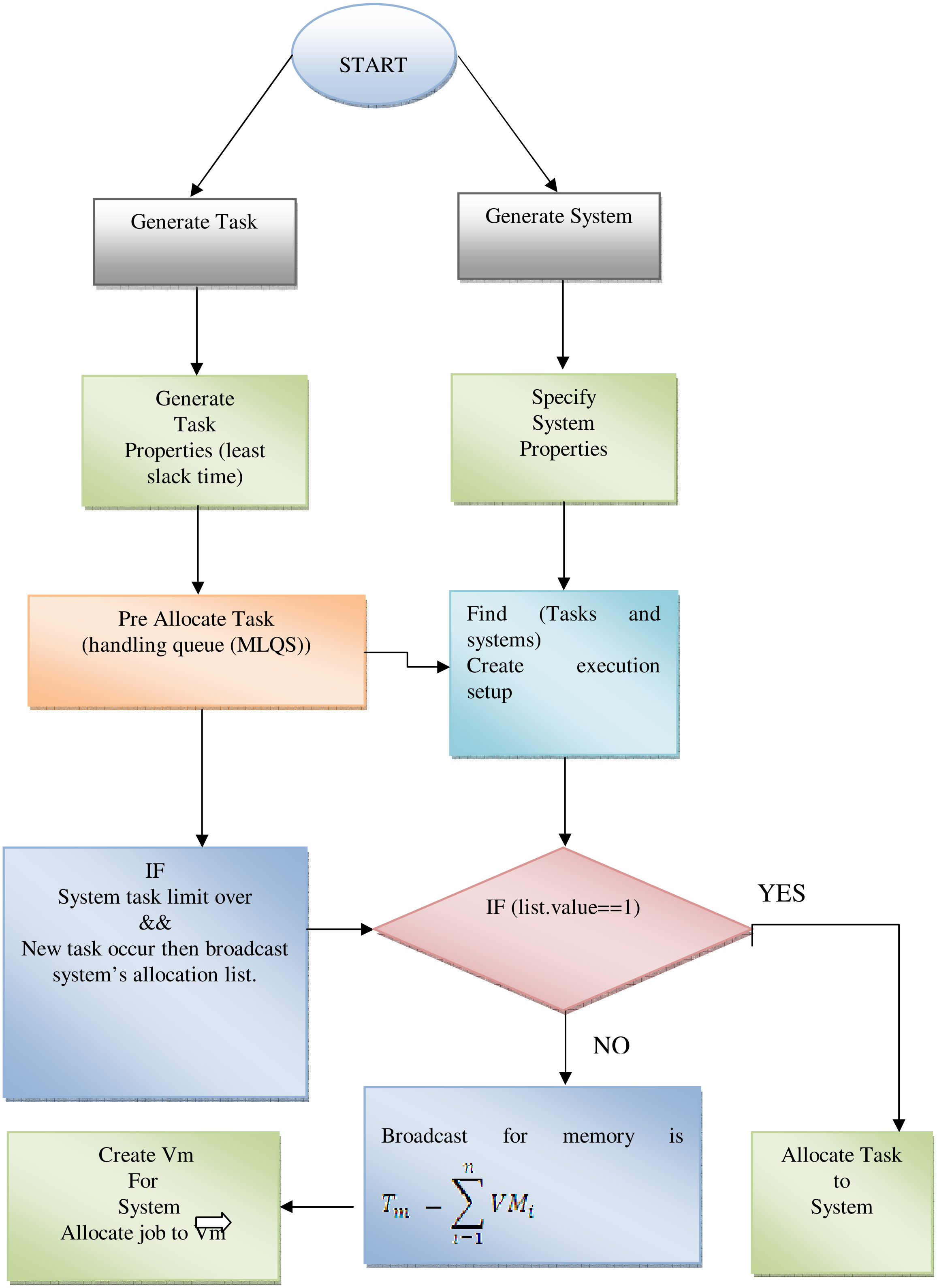

START
Generate Task
Generate System
Generate Task Properties (least slack time)
Specify System Properties
Pre Allocate Task (handling queue (MLQS))
Find (Tasks and systems) Create execution setup
IF System task limit over && New task occur then broadcast system's allocation list.
IF (list.value==1)
YES
NO
Broadcast for memory is
$T_m - \sum_{i-1}^{n} VM_i$
Create Vm For System Allocate job to Vm
Allocate Task to System

Fig 1.1. Flow chart of work done

6. Result Analysis

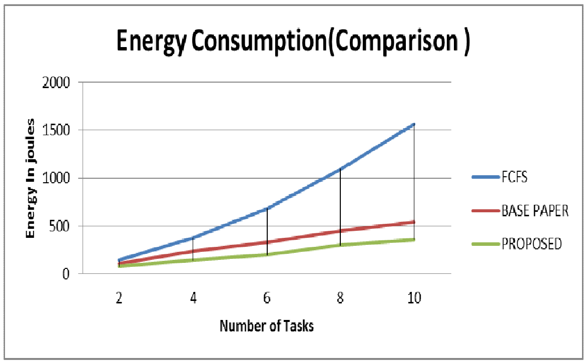
Computation Parameters

- **THRESHOLD: -** Threshold for Least Slack Time algorithm is used to execute tasks make system reliable and boost up speed of execution with minimum energy consumption and take less time to execute them. Multilevel queue scheduling algorithm is used for handling job queues on run time for make system responsive to all their users who send their processes for execution.
- **Energy consumption** = (waiting time of execution * task ideal energy) + task energy.
- **Time**: It is the total amount of time consumed during execution.

Comparison of Energy Consumption:

Tasks	**2**	**4**	**6**	**8**	**10**
FCFS	144	381	685	1097	1565
Base Paper	112	237	329	449	543
Proposed Algorithm	82	197	199	279	363

In this table shows the number of tasks executed by FCFS, Base Paper and Proposed algorithms. And the results shows how much energy consumed by these algorithms. From this research shows that the proposed algorithm consumed less energy from FCFS and base paper algorithms.



The graph shows results of various algorithms used for comparison in this research. The results produced by proposed algorithm shown better then pervious and scheduling algorithm FCFS on same dataset for these entire algorithm. The systems remain same for execution of these tasks but the tasks count on every execution changed due to check and compare the efficiency of these algorithms. In all these condition our proposed approach generate better results and save much more energy than other algorithms as shown in graph on same data set provided to them.

Comparison of Time Consumption:

Tasks	**2**	**4**	**6**	**8**	**10**
FCFS	40	150	290	470	690
Base Paper	30	90	130	170	210
Proposed Algorithm	15	45	80	120	160

In this table shows the number of tasks executed by FCFS, Base Paper and Proposed algorithms. And the result shows how much time is taken by these algorithms to execute tasks. From this research shows that the proposed algorithm take less time for execution from FCFS and base paper algorithm.

7. Conclusion and Future Scope

During the survey of various job scheduling algorithms, it has been widely observed that these algorithms consume vast amount of energy and consume more time for scheduling of jobs. So, the main motive behind this research is to propose an algorithm which consumes less energy and time for scheduling of jobs which are sent to cloud at the same time to access services. Apart from designing the proposed solution, its implementation is also done using Dot Net framework over Microsoft Azure server. Moreover, in order to reflect the optimal efficiency of this composite platform in comparison to their individual counterparts, performance analysis has been done on the basis of energy consumption and time consumption etc. and the outcome is graphically reflected and thoroughly discussed. Thus, from the analysis, we conclude that the proposed algorithm results in more accurate outcomes than the previous scheduling algorithm. In future, the proposed scheduling algorithm can be used with other algorithm to make a hybrid algorithm which may help at the time of scheduling. They would consume less energy and take lesser time for the execution of jobs.

References

[1] Abhishek Kumar Gupta and Kulwinder singh Mann, “Sharing of Medical Information in Cloud Platform”, *Journal of Computer Engineering,* vol. 16, pp. 8-11, March 2014.

[2] Andrew J. Younger, G. Laszewski and L. Wang, “Efficient Resource Management for Cloud Computing Environments”, in *Proceeding of IEEE 10th International Conference on Cloud Computing,* pp. 1-8, 2010.

[3] Antow Beloglazov and R. Buyya, “Energy Efficient Resource Management in Virtualized Cloud Data Centers”, in *Proceeding of IEEE 10th International Conference on Cluster, Cloud and Grid Computing,* pp. 1-2, 2010.

[4] Anubha Jain, Manjj Mishra and Sateesh Kumar Peddoju, “Energy Efficient Computing- Green Cloud Computing”, *IEEE,* 2012.

[5] Amit Garg and C. R. Krishna, “A Review on Scheduling Algorithms in Cloud Computing”, *International journal of Computer Networks and Information Technology,* NITTTR. Chandigarh, 20-21 March, pp. 309-314, 2014.

[6] Arash Delava and Yalda Aryan, “HSGA: A Hybrid Heuristic Algorithm for Workflow Scheduling”, *Springer Conference on Cluster Computing,* pp. 1-9, 2013.

[7] Bo Li, Jinpeng Huai, Jianxin Li and Tianyu Wo, and “An Energy-saving Application Live Placement Approach for Cloud Computing Environments”, IEEE International Conference on Cloud Computing, pp. 17-24, 2009.

[8] Charles Lefurgy and Karthich Rajamani, “On Evaluating Request Distribution Schemes for Saving Energy in Server Clusters”, *IEEE International Symposium on Performance Analysis of Systems and Software,* 2003.

[9] J. Fontan, L. Gonzalez, M. Llorente, R. Montero and T. Vazquez, “Open NEbula: The Open Source Virtual Machine Manager for Cluster Computing”, in *Proceeding of IEEE Conference on Open Source Grid and Cluster Software,* San Francisco, CA, USA, May 2008.

[10] Chee Shin Yeo, Rajkumar Buyya, and Srikumar Venugopal, “Market-oriented Cloud Computing: Vision, Type and Reality for Delivering IT Services as Computing Utilities”, *IEEE,* pp. 13, Sep. 2008.

[11] Che-Lon Hung, Hsiao Wang and Yu Chen Hu, “Efficient Load Balancing Algorithm for Cloud Computing Network”, [Online]. Available: *onlinepresent.org/proceedings,* vol. 2, pp. 251-253, 2012.

[12] Chihyun Jung, Detlef Pabst, Myoungsoo Ham, Marcel Stehli and Marcel Rothe, “An Effective Problem Decomposition method for Scheduling of Diffusion Processes Based on Mixed Integer Linear Programming”, *IEEE,* vol.27, no.3, 2014.

[13] Ching Hsienn Hsu and Tai Lung Chen, “Adaptive Scheduling based on QoS in Heterogeneous Environment”, *IEEE,* 2010.

[14] Chuliang Weng, Minglu Li, Xinda Lu and Z. Wang, “The Hybrid Scheduling Framework for Virtual Machine Systems”, in *Proceeding of Conf. VEE09,* pp. 113-120, 2009.

[15] Dipti Bhansali, Jaee Bansiwal, Juhi Kshirsagar and Shalmali Ambike, “An Optimistic Differentiated Job Scheduling System for Cloud Computing”, *International Journal of Engineering Research and Applications,* vol. 2, no. 2, pp. 1212-1214, 2012.

[16] D. P. Agarwal and R. Bajaj, “Improving Scheduling of Tasks in a Heterogeneous Environment”, in *Proceeding of IEEE Transactions on Parallel and Distributed Systems*, pp. 107-118, 2004.

[17] Dr. Amit Agarwal and Saloni Jain, “Efficient Optimal Algorithm of Task Scheduling in Cloud Computing Environment”, *International Journal of Computer Trends and Technology,* vol. 9, no. 7, pp. 1-6, March 2014.

[18] D. Wang and K. Li, “Cloud Task Scheduling Based on Load Balancing Ant Colony Optimization”, *IEEE International China Grid Conference,* pp. 3-9, 2011.

[19] Fahad Ahmad, M. Saleem Khan, Q. Khan and Sahid Nasser, “Usage & Issues of Cloud Computing Techniques in Small & medium Business Organizations,” *International Journal of Scientific & Engineering Research,* vol. 3, May 2012

[20] Fang Huang, Geoffrey Fox, Gregor Laszewski, Jai Dayal , Lizhe Wang and Tom Frulani, “Task scheduling with ANN-based Temperature Prediction in a Data Center: A Simulation-based Study”, *Springer Journal of Engineering with Computers ,* vol. 27, no. 4, pp. 381-391, February 2011.

[21] F. maurer, J. Ferrira and Y. Ghanm “Emerging Issues and Challenges in Cloud Computing -A Hybrid Approach”, *Journal of Software Engineering and Application,* vol. 5, no. 11A, pp. 923-937, 2012.

[22] Frank Wang and G. Akanmu, “Min-Min Scheduling Algorithm for Heterogeneous Cloud Servers”, *International Journal of Computer Engineering,* vol. 2, no. 4, pp. 15-16, 2012.

[23] Frank Wang, G. Akanmu, H. Chen and N. Helian “User-Priority Guided Multiple Min-Min Scheduling Strategy for Load Balancing in Cloud Computing”, *IEEE,* pp. 1-8, 2013.

[24] Gan Guo Ning and Hrang Ting Lei, "Genetic Simulated Annealing Algorithm for Task Scheduling based on Cloud Computing Environment," In *Proceedings of International Conference on Intelligent Computing and Integrated Systems,* pp. 60-63, 2010.

[25] Gau Zohong Wen and Zhang Kai, "The Research on Cloud Computing Resource Scheduling Method Based on Time Cost-Trust Model", *IEEE,* pp. 939-942, 2012.

[26] Gregorvon Laszewski, Andrew J. Younge and Lizhe Wang, "Efficient Resource Management for Cloud Computing Environments", in *Proceeding of IEEE 10th International Conference on Cloud Computing,* pp. 1-8, 2010.

[27] G. Sadhasivam and S. Selvaranil and, "Improve Cost-Based Algorithm for Task Scheduling in Cloud Computing", *IEEE,* pp. 1-5, 2010.

[28] Haiyang Wang , X. M. Cui, and Y. Bi, "A Multiple Qos Constrained Scheduling Strategy of Multiple Workflows for Cloud Computing", in *Proceeding of IEEE International Conference on Parallel and Distributed Processing with Application,* pp. 629-634, 2009.

[29] Hiroshi Nakamura, Naohiko Mori, Satoshi Itoh, Satoshi Sekiguchi, T. Shimizu and Yuetsu Kodama, "Imbalance of CPU Temperature in a Blade System and Its Impact for Power Consumption of Fans", *in Proceeding of IEEE International Conference on Green Computing,* pp. 81-87, 2011.

[30] http://en.wikipedia.org/wiki/Cloud_computing/Deployment_models.

[31] http://en.wikipedia.org/wiki/Visual_Basic_.NET.

[32] Isam Azawi Mohialdeen, "Comparative Study of Scheduling in Cloud Computing Environment", *Journal of Trends Computer Science,* vol. 9, pp. 252-263, 2013.

[33] J. Fontan, L. Gonzalez, M. Llorente, R. Montero and T. Vazquez "Open NEbula: The Open Source Virtual Machine Manager for Cluster Computing", in *Proceeding of IEEE Conference on Open Source Grid and Cluster Software,* San Francisco, CA, USA, May 2008.

[34] Jiandum Li, Junjie Peng, Wo Zhang and Zhou Lie, "An Energy Efficient Scheduling Approach Based on Private Cloud", *Journal of Information and Computational Science,* vol. 8 , no. 4 , pp. 716-724, 2011.

[35] Jiayin. Li, M. Qiu, X. Qin, "Feedback Dynamic Algorithms for Preemptable Job Scheduling in Cloud Systems", *IEEE,* 2010.

[36] Lizhe Wang and Sameer Khan, "Review of Performance Metrics for Green Data Centers: A Taxonomy Study", *The Journal Super Computing,* vol. 63, no. 11, pp. 639-656, March 2003.

[37] Lucio Grandinetti, Manjot Devare and M. Sheikhalishahi, "A General purpose And Multi-level Scheduling Approach in Energy Efficient Algorithm", in *Proceeding of CLOSER Conference,* 2011.

[38] Luna Mingyi Zhang, L. Keqin, and Y. Q. Zhang, "Green Task Scheduling Algorithm With Speed Optimization on Heterogeneous Cloud Servers", *IEEE,* 2010.

[39] Mahammad Sajid and Zahid Raza, "Cloud Computing Issues and Challenges," in *Proceeding of IEEE International Conference on Cloud, Big Data and Trust,* pp. 35-41, 2013.

[40] Margret Johnson and R. A. Preethima, "Survey on Optimization Techniques for Task Scheduling in Cloud Computing," *International Journal of Advanced Research in Computer Science and Software Engineering,* vol. 3, no. 12, pp.413-415, 2013.

[41] Monica Gahlawat and P. Sharma, "Analysis and Performance Assessment of CPU Scheduling Algorithm in Cloud Sim," *International Journal of Applied Information System (IJAIS),* vol. 5, no. 9, July 2013.

[42] Neetika gupta, Jyoti Kataria and Abhay Bansal, "A Survey on Cloud Providers and Migration Issues", *International Journal of Computer Applications,* vol. 56, no.14, pp. 38-43, October 2012.

[43] "NIST Definition of Cloud Computing," vol. 15, csrc.nist.gov/groups/SNS/cloud computing/cloud- def v15.doc.

[44] Pawar, C.S and R. B, "Priority Based Dynamic resource allocation in Cloud computing", *IEEE International Symposium on Cloud and Services Computing,* pp. 1-6, 2012.

[45] R. Suchithra, "Efficient Migration –A Leading Solution for Server Consolidation," *International Journal of Computer Applications,* vol. 60, no.18, December 2012.