Volume: 10 Issue: 05 Pages: 4013-4018 (2019) ISSN: 0975-0290

Energy Efficient Scheduling Algorithm for Cloud Computing Systems Based on Prediction Model

*G. Prasad Babu
Research Scholar, University of Technology, Jaipur.
Email: garikapatiprasadbabu.phd.uot@gmail.com
Dr. A. K. Tiwari
Associate Professor, Department of C.S.E, University of Technology, Jaipur

----------ABSTRACT----------

Existing cloud resource scheduling approaches have mainly concentrated on enhancing the reducing power consumption and resource utilization by enhancing the legacy heuristic algorithms. Although, different resource-intensive applications running on cloud data centers in realistic scenarios have significant results on the power consumption and cloud application performance. Furthermore, occurring peak loads may lead to a scheduling error, which can significantly effects on the energy efficiency of scheduling algorithms. At peak loads may lead to scheduling errors because there is no prediction model to predict the coming resource utilization of a data center through the data collected by the monitoring model. Effective scheduling mechanism gives an optimal solutions for complex problems while providing the Quality-of-Service (QoS) and avoiding Service Level Agreement (SLA) violations. To enhance the resource scheduling mechanism in cloud environment, predicting future workload to the each virtual machine pool in different manners like number of physical machines, number of virtual machines, number of requests and resource utilization etc., is an essential step. According to the prediction results, resource scheduling can be done in the right time, while avoiding QoS dropping and SLA violations. To achieve efficient resource scheduling, proposed approach lease advantages of prediction models. The proposed algorithm consists of a prediction model which is based on iterative fractal model and a scheduler which is based on an improved heuristic algorithms. Proposed scheduler algorithm is responsible for scheduling of resources while reducing the energy consumption and giving the guaranteeing the QoS.

Keywords - Cloud computing, Energy efficient, Prediction model, Scheduling algorithm, Virtual machine.

Date of Submission: Jan 28, 2019 Date of Acceptance: Apr 22, 2019

I. INTRODUCTION

In distributed computing, cloud computing has been developing vastly for the past few years, towards achieving technical improvements in distributed computing. Cloud computing is one of the trending models that has progressed from adopting virtualization technology, utility computing and service oriented architectures. Cloud computing providing many services such as data storage services, web applications and network structures that could be allotted and departure with less effort of cloud owner management. In cloud computing services, Infrastructure as a service (IaaS) is providing the services by deploying the virtual machines in cloud data centers. Usually, IaaS allocate the virtual machines with the help of scheduling policies to the cloud users. For example, round robin (RR) scheduling algorithm allocate the virtual machines based on the cloud user requests. Weighted least scheduling algorithm allocate the weight to each virtual machine and highest weight value of virtual machine is allocated to more number of requests. Rank scheduling policy also allocate the ranks to virtual machines then assigned the virtual machines according to the request rate.

In recent years, resource scheduling has become one of the challenging task in cloud based industries. Most of the reasons are related to the unexpected incoming workload and insufficient resources issues. In this case, centralized management performs the resource scheduling to allocate the resources to users. The centralized management maintains the status of the virtual machine and scheduling policy in each intra cloud and allocating the requests to the virtual machines. Heavy workloads may lead to scheduling errors because there is no prediction model to predict the coming resource utilization of a data center. Existing scheduling mechanism could not provide an optimal solutions for complex problems to providing the Quality-of-Service (QoS) and avoiding the Service Level Agreement (SLA) violations.

A proposed research direction, which applies modern prediction model and heuristic algorithms to scheduling on cloud computing will give efficient results. The proposed method is a combination of prediction model and heuristic model to leverage their strengths for scheduling. The basic idea of proposed method is predicting future workload to the each virtual machine pool in different manners like number of physical machines, number of virtual machines, number of requests and resource utilization etc., and according to the prediction results, resource scheduling can be done in the right time. The proposed method is responsible for providing efficient scheduler module for cloud virtual machine to avoid the SLA violations while providing QoS. The major contribution of the paper can be summarized as 1) Development of a scheduling algorithm based on prediction model. 2) Simulation of the proposed algorithm in benchmark and synthetic datasets. 3)

Volume: 10 Issue: 05 Pages: 4013-4018 (2019) ISSN: 0975-0290

Comparison of the experimental results with the existing algorithms.

The rest of the paper structured as follows. In Section 2, presents related work and motivation of resource scheduling algorithm. Section 3 presents, proposed scheduling approach. Overview of experimental and evaluation of proposed method results are presented in Section 4. Finally, Section 5 presents, conclusion of this paper.

II. BACKGROUND AND RELATED WORK

2.1 Scheduling Algorithm Role in Cloud Computing

The scheduling solution can be defined as: For a given set of machines M = {M_1, M_2, M_3… M_n}, finding an optimal solution to schedule a given set of tasks T = {T_1, T_2, T_3… T_n}. In cloud computing system, the scheduling problem can be defined as the workflow problem which is divided into two categories: task level scheduling and service level scheduling. A task level scheduling can be performed at unified resource layer and service level scheduling can be performed at platform layer. For example, in cloud scheduling problems were solved by directed acyclic graph (DAG). The main idea of the DAG is, nodes of the graph represents set of tasks and edges represents the dependencies between the tasks. Then, scheduling problem can be defined as follows:

$$\text{Minimize } f(s) = C_{max}(s) + \sum_{i=1}^{m} Ci + \sum_{j=1}^{n} Cj \quad (1)$$

In equation 1, f(s) is the target function, $C_{max}(s)$ is the completion time of the last task, m represents the number of machines, n represents the number of tasks and C_i, C_j represents the cost of processing the i^{th} task on the j^{th} machine. To measure the solution for scheduling algorithms, make span $C_{max}(s)$ is the main parameter. Additionally maximum flowtime, maximum lateness and maximum tardiness can also use for scheduling measure parameters. Following parameters are defined from literature work for measuring the efficiency of a scheduling algorithm in cloud computing environment.

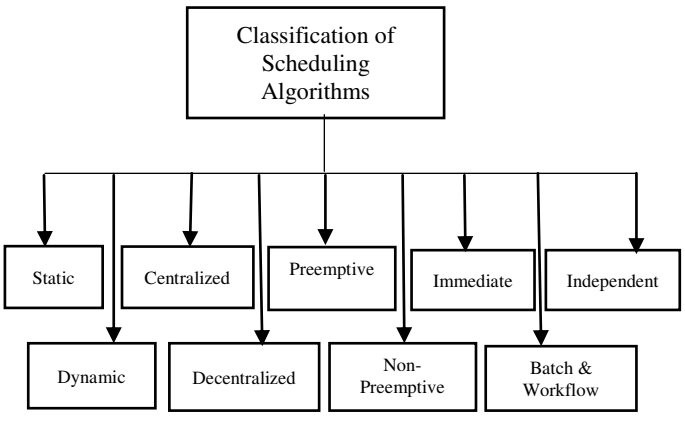

Fig 1:Taxonomy of scheduling algorithm

i) Reliability: The scheduling algorithm should be reliable because while transferring task from one location to other leads to increased waiting time. When waiting time increase, cloud user face the dissatisfaction.

ii) Adaptability: The scheduling algorithm should be capable of adapting the dynamically changing user requests and provide task scheduling in minimal amount of time.

iii) Fault Tolerance: The scheduling algorithm should be ensure fault tolerance, the completion of problem must be minimum amount of time.

iv) Throughput: The algorithm must ensure maximum throughput at a minimal expenses. If a scheduling algorithm doesn't reach maximum throughput, the algorithm gives maximum processing time.

v) Makespan: The scheduling algorithm should be minimize the waiting time of task to allocate a resource.

In cloud computing, heuristic scheduling algorithms can be used to provide better scheduling models rather than rule based scheduling algorithms. Figure 1, represents the basic model of heuristic scheduling algorithm with three important operators – transition, evaluation and determination. These three operators are used for searching for optimal solution on convergence process. Here, n denotes the iteration number and n_{max} represents the maximum number of iterations. The transition operator creates the solution by using constructive method, the evaluation operator measures the fitness of defined solution and determination operator defines the further search directions based on the defined the solution from the transition and evaluation operators.

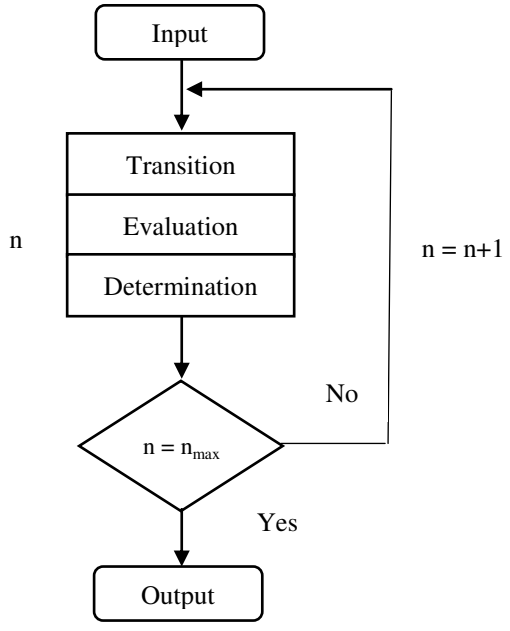

Fig 2: Workflow of basic heuristic scheduling algorithm

Volume: 10 Issue: 05 Pages: 4013-4018 (2019) ISSN: 0975-0290

2.2 Classification of Scheduling Algorithms

Figure 1, represents the taxonomy of scheduling algorithms in cloud environments. These are various kinds of scheduling algorithms which are useful to cloud providers to schedule the resources to the cloud application with in a time. These scheduling algorithms helps to increase the efficiency of cloud environment.

a) Static and Dynamic Scheduling

In Static Scheduling tasks are scheduled at a time. Information about all the tasks in cloud application is pre assumed to be applicable by the time of cloud resources is scheduled and also it is predicted that cloud resources are estimated available all the time and there is no tasks failures. Dynamic scheduling tasks are dynamically applicable for scheduling over the all-time by the scheduler with no errors, to able of estimating completion time in an advance. The dynamic scheduling algorithms concerns job priority and resource failures. When the cloud resource fails execution will be stop. To avoid this issue scheduler algorithm adopts the tasks to another resource. When job with higher priority comes into the queue in a preemptive selection, dynamic scheduler algorithm allocate a resource to the task or job. If needed virtual resource is busy and it stops the task executing and allocates it to the new job with higher priority.

b) Centralized and Decentralized Scheduling

Centralized scheduling and decentralized scheduling contrast in the available of the cloud resources and information of the complete system. In centralized scheduling algorithms have more information of resources. This scheduler has information of the complete system. It monitor the resources continuously. The advantages of the centralized scheduling algorithm is, implementation. Distributed or decentralized scheduling algorithm has no centralized scheduling. The decisions of scheduling, are shared by different schedulers and has less complexity than centralized schedulers.

c) Pre-emptive and Non-Preemptive Scheduling

Preemptive scheduling algorithm is the scheduling constraints allows each task to be interrupted during execution and a task can be shifted to other resources leaving its originally allocated resources unused to be available for other tasks. It takes the priority of the task. Non-Preemptive scheduling algorithms does not preempt the cloud resource until tasks completes its execution.

d) Immediate and Batch Mode Scheduling

In Immediate scheduling algorithms, scheduler allocates tasks as soon as it come. In this process on available cloud resources there is no waiting time. In Batch mode scheduling algorithms, the Scheduler holds arriving tasks as group of tasks to be allocated over time intervals. This process is considered to be simple as it map tasks to available resources based its requirements.

e) Independent and Workflow Scheduling

Independent scheduling algorithms completes the jobs independently. In Workflow scheduling algorithms there exists dependency between the jobs. Dependency means there are priority orders exiting among jobs. A sub job cannot execute until its main job completes the execution. Workflow of jobs are presented by Directed Acyclic Graph (DAG) notation. Each job can start its execution only when all priority jobs in DAG are gets completed.

2.3 Related Work

There have been a number of scheduling algorithms for energy efficiency in cloud computing. Leverich et al. [1] addressed the solution for selection of physical machines in a cluster to execute the tasks while satisfying the power consumption parameter. The proposed method approaching the scheduling model to select a physical machine in each cluster. After collecting the data from each cluster, proposed model is working on scheduling approach to schedule the physical machine to each cluster. Virtualization is the one of the major technique in cloud computing to utilize the resources fully. Various virtual machine scheduling algorithms [2] have been developed to dynamically allocate the resources to cloud users. These allocation algorithms are classified into two categories, in first step virtual machines will be allocating onto physical machines and next step is, assigning physical machines to virtual machines. Lang et al. [3] proposed a mechanism to run the all workloads simultaneously in all physical machines to save more energy based on the incoming workloads with the help of scheduling approach. The proposed mechanism working for mapping and allocation problem to reduce the power consumption due to virtual machine workload. The DVFS (dynamic voltage frequency scaling) is the main technique in cloud computing to reduce the power consumption. In datacenters, the DVFS technique mainly used for makes trade-offs between performance and processor power to optimize the power consumption. Wang et al. [4] addressed the solution for energy efficient scheduling polices with the help of DVFS technique to reduce the CPU frequency. So that, the proposed scheduling technique reducing the carbon emission and increasing the revenues of cloud provider. Wu and Li [5] introduced the RIRA (Relaxation Iterative Rounding Algorithm) algorithm with the help of DVFS technique. This mainly concentrated on minimizing the energy consumption. Watanabe et al. [6] proposed scheduling algorithm to reduce the energy consumption in clusters. The proposed algorithm presented a scheduling approach for resources allocation. K.D. Kumar et al. [7] presents the survey in the view of machine learning algorithm roles in cloud computing environments. Authors explained about machine learning algorithms and their roles in cloud computing environments. Each algorithm used for predicting the resources to the cloud applications. Predicting the resources to the cloud applications is the very important task for cloud provider. At the same time, predicting accuracy results also challenging task. But, accurate algorithms increase the efficiency of cloud environments. With accurate prediction results, cloud provider can schedule the resources to the cloud application in an advance. S. Mohamed et al. [8] proposed the method to optimize the power consumption which is

Volume: 10 Issue: 05 Pages: 4013-4018 (2019) ISSN: 0975-0290

arising in cloud datacenters maintenances. Authors focused on the reduction of power consumption. Usually datacenters produces heavy Co_2 and which can lead pollution of environments. Authors proposed 'green cloud' approach to reduce the power consumption. B. Dinh et al. [9] addressed the solution to reduce the power consumption with the help of prediction algorithms. Authors proved, with the help of machine learning algorithms cloud environments can increase the efficiency of optimization. The proposed lgorithm addressed the solution for energy efficiency. A.T. Makaratzis et al. [10] proposed algorithm for energy model. The whole work developed in simulation tool. Author addressed the simulation framework to solve the power consumption problem. The simulation results proves, effective scheduling algorithm gives the solution for power consumption problems. D. Mehiar et al. [11] proposed an energy efficient algorithm for resource allocation in cloud. This algorithm has taken the advantages of scheduling algorithm to allocate the resource to the cloud applications. Z. Wei et al. [12] proposed a three dimensional virtual scheduling algorithm to reduce the power consumption. The main idea of the proposed system is, using scheduling algorithm with three parameters to allocate the right amount of resources.

III. PROPOSED SCHEDULING APPROACH

Motivated by the issues in many existing scheduling algorithms, the proposed algorithm presents a new scheduling algorithms to reduce the energy consumption. The proposed approach investigated the source of energy consumption in the cloud environment while considering the heterogeneity computing environment. Additionally, the proposed approach used a prediction method [7] which is based on fractal method to assist the scheduling algorithm make proper decisions to allocate the resources.

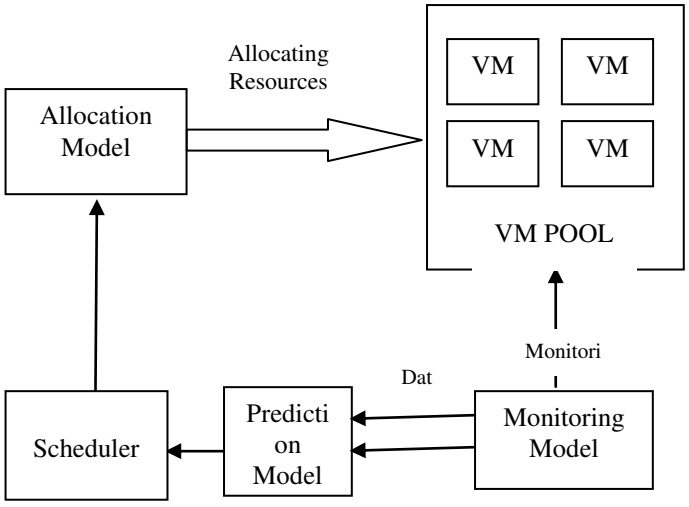

Fig 3: Architecture of the Proposed Scheduler Model

The proposed scheduling approach as shown in the figure 2, presenting four modules. First, monitoring model collects resource workloads from every physical machines in each cluster like disk utilization, memory utilization and CPU utilization. The monitoring model will send this data to prediction model to analyze the utilization of resources. The prediction model analyze the utilization records then send the results to scheduler model. The scheduler model is responsible for allocating resources to particular hosts based on the prediction model results. Finally, the allocation model allocates the resources for the host. The main role of the allocation model is allocating a suitable virtual machine to incoming requests from cloud users. From many existing algorithms found that, in resources CPU load has efficient impact and strong correlation for research purpose. Motivated by this point, proposed method used to predict CPU utilization based on iterative fractal method. The time has taken as a series of time points with intervals like t_1, t_2, $t_{3...}$ t_n and predict the usage of CPU in the cluster like P_1, P_2, P_3… P_n. From the results, the proposed method prediction values has the same number of values as the real time values after running two prediction steps. The equation 2 represents the Pearson correlation coefficient to find the similarity between predicted CPU load value and real CPU load. The correlation between two values is calculated as follows.

$$r = \frac{\sum XY - \frac{\sum X \sum Y}{N}}{\sqrt{(\sum X^2 - \frac{(\sum X)^2}{N})(\sum Y^2 - \frac{(\sum Y)^2}{N})}} \quad (2)$$

As shown in the Pseudocode, the monitoring model gather the information from virtual machine pool at interpolation points. Iterative fractal prediction runs for every five minutes to calculate the CPU utilization of virtual machines. The IFM (iterative fractal method) generates the similar workload point values with comparison of current value (C-CPU) and predicted value (P-CPU). If a similar workload value is closer to the previous workload value, the result of correlation factor is bigger. Based on correlation factor, scheduler model schedule the resources to particular cloud user with the help of allocation model.

IV. IMPLEMENTATION & RESULTS

4.1 Simulation Scenario

The proposed scheduling algorithm implemented and evaluated by using simulation environment. The proposed model implemented in Cloudsim simulation environment with the references of existed scheduling algorithm in the simulation environment. In this experiment, datacenter is consisted 100 physical machines. The results are compared with Round Robin (RR) scheduling algorithm and Minimum Migration Time (MMT) scheduling algorithm, First-Fit (FF) scheduling algorithm by using Google trace dataset. In Round robin algorithm, attribute all the incoming tasks to the available virtual machines and each virtual machine gives a quantum of time to each assigned task to be executed. If task execution has not finished, the process is interrupted and the next task execution on this specific VM list will take place. In MMT scheduling algorithm, the scheduler verifies the migration

Volume: 10 Issue: 05 Pages: 4013-4018 (2019) ISSN: 0975-0290

time of all virtual machine which are placed in host. The main aim of this algorithm is, considering the virtual machine which is having less migration time to migrate the virtual machine from one host to another.

The FF algorithm gives the results of virtual machine to migrate from one host to another host. This algorithm identifies the suitable cloud resource to execute the jobs efficiently. The simulation process has considered for one day to predict the results 24 hours before and CPU utilization low and high threshold values are considered 0.2 and 0.8 respectively. The monitoring model time interval is 5 minutes and the IFM factor weightage values is 0.6 and 0.3.

Pseudocode: Proposed Model

```
Begin
Monitoring of the CPU utilization of VM's
At each point of time't', gathered new resources data.
While (t < 300) (time interval is 300 seconds)
 // IterariveFractalPrediction
   Calculate the CPU utilization value for each VM
   IFM = comparisonProcess(C-CPU, P-CPU)
// by using Pearson correlation
   k = 0
   While k! = P do
     getPredictedValue(k)
     + + k
   end while
Compare the results with new resources data.
Send the details to scheduler model
Based on the scheduling results allocate the resources
End
```

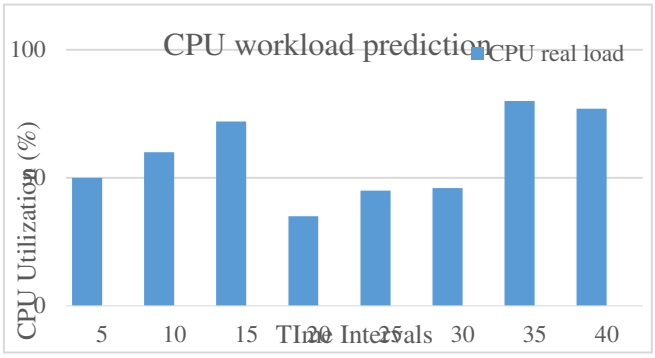

Fig 4: CPU workload prediction

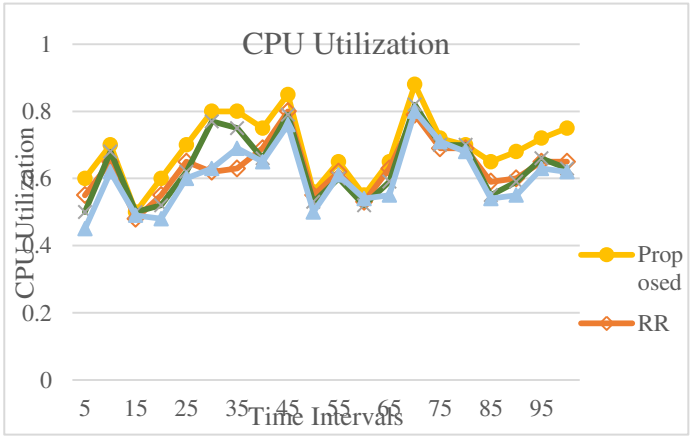

Fig 5: CPU utilization at time intervals

Volume: 10 Issue: 05 Pages: 4013-4018 (2019) ISSN: 0975-0290

Table 1: Average CPU utilization for Google trace dataset

	Proposed	RR	MMT	FF
Minimum CPU utilization	0.59	0.56	0.49	0.53
Average CPU utilization	0.72	0.66	0.69	0.60
Maximum CPU utilization	0.86	0.80	0.81	0.75

4.2 Result Analysis

The simulation process runs until the simulation time expires. The following results presents, proposed algorithm gives the better results than RR, MMT and FF algorithms, does means proposed algorithm has a good impact on reducing power consumption. The figure 4 represents, comparison between CPU real workload and CPU predicted workload. The proposed method, predicted workload values which are closer to CPU real workload values from Google trace dataset. The figure 5 represents CPU utilization of proposed method, round robin, minimum migration time and first fit algorithm values. The proposed method has utilized more CPU memory than remaining algorithms and results has proved, proposed method gives more profits while reducing the power consumption of CPU resources.

V. CONCLUSION

In this paper, the proposed method energy efficient scheduling algorithm based on prediction model used for scheduling the resources in time. Mainly, proposed algorithm has taken the leverage of prediction methods which is working based iterative fraction method. The CPU utilization is the primary factor in the part of power consumption of cloud resources. In experiments, simulation model used real traces of Google clusters. The proposed approach results presents, proposed algorithm has utilized more CPU memory than remaining algorithms which are round robin, minimum migration time and first fit.

REFERENCES

[1] Leverich, C. Kozyrakis, On the energy (in) efficiency of Hadoop clusters, *Operating System Review* **44** (1) (2010) 61–65.

[2] Zhen Xiao, Weijia Song, Qi Chen, Dynamic resource allocation using virtual machines for cloud computing environment, *IEEE Transaction on Parallel Distributed Systems* **24** (6) (2013) 1107–1117.

[3] W. Lang, J.M. Patel, Energy management for MapReduce clusters, *PVLDB* **3** (1–2) (2010) 129–139.

[4] L. Wang, S. U. Khan, D. Chen, J. Kolodziej, R. Ranjan, C.-z. Xu, A. Zomaya, Energy-aware parallel task scheduling in a cluster, *Future Generation Computer Systems* **29** (7) (2013) 1661–1670.

[5] A.M. Chirkin et al., Execution time estimation for workflow scheduling, *Future Generation Computer Systems* **75** (2017), 376-387.

[6] Wu and Li, Energy-aware scheduling for frame-based tasks on heterogeneous multiprocessor platforms. In: IEEE 41st *International Conference on Parallel Processing (ICPP)*. IEEE, 430–439.

[7] K.D. Kumar et al., Prediction methods for effective resource provisioning in cloud computing: A survey, *Multiagent and Grid Systems* **14** (3) (2018), 283-305.

[8] S. Mohamed and A. Shami, An evergreen cloud: Optimizing energy efficiency in heterogeneous cloud computing architectures, *Vehicular Communications* **9** (2017), 199-210.

[9] B. Dinh et al., Energy efficiency for cloud computing system based on predictive optimization, *Journal of Parallel and Distributed Computing* **102** (2017), 103-114.

[10] A.T. Makaratzis, M.G. Konstantinos and D. Tzovaras, Energy modeling in cloud simulation frameworks, *Future Generation Computer Systems* **79** (2018), 715-725.

[11] D. Mehiar et al., Energy-efficient resource allocation and provisioning framework for cloud data centers, *IEEE Transactions on Network and Service Management* **12** (3) (2015), 377-391.

[12] Z. Wei, Y. Zhuang and L. Zhang, A three-dimensional virtual resource scheduling method for energy saving in cloud computing, *Future Generation Computer Systems* **69** (2017), 66-74.

[13] E. Jafarnejad, A.M. Rahmani, Ghomi and N.N. Qader, Load-balancing Algorithms in Cloud Computing: A Survey, *Journal of Network and Computer Applications* **88** (2017), 50-71.

[14] A.M. Sadegh, M. Ghobaei and A.N. Toosi, Auto-scaling web applications in clouds: a cost-aware approach, *Journal of Network and Computer Applications* **95** (2017), 26-41.

[15] V.J. Luis et al., SaaS enabled admission control for MCMC simulation in cloud computing infrastructures, *Computer Physics Communications* **211** (2017), 88-97.

[16] Krunal N. Vaghela et al., Job Scheduling Heuristics and Simulation Tools in Cloud Computing Environment: A Survey, *International Journal Advanced Networking and Applications*, **10** (2), (2018), 3782- 3787.

[17] G. RamaSubba Reddy, Optimal Resource Allocation and Reservation using DAR in Large Scale Applications, *International Journal Advanced Networking and Applications*, **10** (2), (2018), 3822-3828.

[18] Rupinder Kaur et al., Efficient Task Scheduling using Load Balancing in Cloud Computing, *International Journal Advanced Networking and Applications*, **10** (3), (2018),3888-3892.