Real Time Multi Factor Energy Approximation based Scheduling for Improved QoS Performance in Cloud

Mithun D'Souza, Research Scholar, Bharathiar University, Coimbatore.

Assistant Professor, Department of Computer Science, PES Institute of Advanced Management Studies, Shivamogga, Karnataka, India. E-mail: mitundsouza@gmail.com

Dr.S. Sathyanarayana, Srikantha First Grade College for Women, Mysore, Karnataka, India. E-mail: ssn_mys@yahoo.com

Abstract--- The problem of scheduling cloud resources has been well studied. There are number of traditional techniques exist to perform scheduling, but suffer with higher energy consumption and suffer to achieve QoS performance. To solve this issue, an efficient real time multi factor energy approximation based scheduling scheme is presented in this paper. The method classifies the resources according to their properties like energy consumption, task support and so on. For any task submitted, the method performs the selection of virtual machine according to the underlying resources. For any resource available in the cloud, the method computes different task support measures (TSM) for various resources like CPU, Memory and Bandwidth. Also, for the submitted task, the method computes the task requirement support towards various cloud resources. Using the task support measure, the method performs energy approximation for the resources. Using both the measures, the method performs scheduling of resources according to the requirement support value. The proposed scheme improves the performance of scheduling and reduces the energy utilization and consumption.

Keywords--- Cloud Computing, QoS, Scheduling, Energy Approximation, TRS, Resource Utilization.

I. Introduction

The growth of information technology supports the organizations in performing various tasks by accessing different services. The environment provides different services like SaaS (Software as a Service), Infrastructure as a Service (IaaS) and PaaS (Platform as a Service). However, the IaaS has the great impact in the organization which attracts the organization towards cloud environment. The job set available with any organization needs specific set of resources with set of properties. Some of the jobs or a task requires high configuration resources. Any processing machine contains CPU, memory and bandwidth capacity. Not all the tasks require all the resources to be at specific level. The requirement varies according to the task submitted.

Scheduling is the process of allocating different resources to the task being submitted. The user will be submitting number of task to the system where the scheduler has to identify the task and their requirement. According to the required resources identified, they must be scheduled or allocated to the task to complete them. There are number of scheduling strategy available some of them use priority, and some of them uses the first come first basis and few uses the time of completion and so on. Whatever the scheduling strategy being used, the ultimate aim is to maximize the performance of all the factors of QoS. The QoS performance of cloud environment greatly depending on various parameters like energy, resource utilization, idle time minimization, and reduction of makespan time. By considering all these factors the quality of service of cloud environment can be improved.

In general, the cloud contains number of resources which can be accessed by the tasks being submitted. Each resource, consumes set of energy or power to perform the specific task. Using higher capacity resource to a small task really affects the performance of entire system. The performance of environment depending on how the resources are utilized and how the idle times of the resources are reduced. To improve the performance there are number of allocation and scheduling algorithm available which uses different features namely execution time, energy, bandwidth, memory capacity and so on. However, the methods suffer to achieve higher performance in throughput, energy consumption and so on.

The scheduling of resources should be performed according to the requirement of resources for the task given. By giving the required resources to the task the performance can be improved. This increases the requirement of scheduling the virtual machines over the resources available in efficient manner. This research is about scheduling the virtual machines over the resources available in such a way to reduce the energy consumption. By considering

DOI: 10.5373/JARDCS/V12I8/20202466
ISSN 1943-023X

Received: 16 June 2020/Accepted: 18 July 2020

all these, this paper presents a multi factor energy approximation based scheduling algorithm towards QoS development in cloud. The method collects the requirement of different tasks, and performs energy approximation. Further, the task requirement support is measured to perform scheduling of task in the platform. The detailed approach is presented in the next section.

II. Related Works

Various approaches are discussed towards the problem of scheduling in cloud. Such methods are discussed in detail in this section.

In [1], the author present the detailed information on scheduling the task and the challenges present in the problem are described well. Also, the issues of scheduling in cloud environment has been well briefed and analyzed in [2]. In [3], the author presented a heuristic approach which combines analytical hierarchy and bandwidth aware scheduling which also includes longest exempted preemption scheduling approaches.

In [4], the author presented a load based scheduling algorithm named LIF which identifies the integrated loads of different tasks and based on the features the method performs scheduling. In [5], the author presented a scheduling scheme where the resources are very much limited according to the market orient scheme and performs distribution of task among the clients. Similarly, in [6], the same market approach is utilized in service allocation among the participants. It identifies the order in which combination of services can be allocated to perform scheduling.

In [7], the author presented a market orient strategy in scheduling the resources which consider the cost of tasks in accessing the resources. The selection of scheduling sequence is selected according to the task. In [8], a log based approach is presented which consider the pervious logs of scheduling and based on the performance of resources in performing the task assigned, the scheduling is performed.

In [9], the author introduced a randomized online stack-driven booking calculation (ROSA) and hypothetically demonstrate the lower bound of its focused proportion. Our recreation demonstrates that ROSA accomplishes a focused proportion near the hypothetical lower bound under a unique case cost work. Follow driven reenactment utilizing Google group information shows that ROSA is better than the ordinary web based planning calculations as far as cost sparing.

In [10], the author presented a policy based load balancing approach is presented which uses the ant colony to perform optimization to reduce the makespan. Similarly in [11], an modified ACO technique is presented to improve the performance of scheduling the VM. Similarly in [12], the author presented a PSO based resource renumbering strategy for scheduling the resources of cloud. It considers the cost of resource at each time stamp to perform scheduling. According to the cost incur by any task, the scheduling is performed.

The scientific workflow are used in scheduling the resources in cloud where the services are allocated according to the workflow. The PSO has been used in optimizing the workflow towards performance development [13]. A deadline based cost effective approach is presented in [13], which consider the varying performance of VM and delay incur by the VM in scheduling.

In [15], the author presented a QoS orient scheduling scheme which considers multiple scheduling strategy and according to the QoS requirement, a single strategy has been selected to perform the user task. Dynamic scheduling is always much better which consider the changing behaviors. Such approach is presented in [16], which uses genetic algorithm in the selection and allocation of resources.

A priority based scheduling strategy is presented in [17], which identifies and allocates resources according to the priority of users and the length of task. In [18], the author presented a Master Service Uniform MultiRound scheme which works according to the uniform multi round. The method consider the bandwidth availability in scheduling and improves the performance.

In [19], the author presented a energy based task allocation scheme and resource allocation strategy over Green Sched model. The method consider the deadline, cost and so on in scheduling. In [20], an genetic algorithm based meta heuristic scheduling strategy is presented to reduce the cost of workflow. Also, the method consider the deadline with reduced cost. A PSO based proactive static scheduling scheme is presented which can perform scheduling with preemptive nature [21]. Similarly, a genetic algorithm based model is presented to handle the deadline conditions. According to the conditions a sequence has been selected and optimized for the cost also [22].

In [23], an power orient scheduling scheme is presented which consider the power claim by different resources in scheduling and resource allocation. The method estimates the resource performance according to the execution

DOI: 10.5373/JARDCS/V12I8/20202466
ISSN 1943-023X

Received: 16 June 2020/Accepted: 18 July 2020

time and power claim. Similarly, in [24], an dynamic power-saving resource allocation (DPRA) strategy is presented which uses the PSO in optimizing the selection of resource which uses energy as the optimization factor.

An integrated scheme of scheduling is presented which uses energy as the important factor. The method considers the memory and bandwidth the resource has in estimating the strength of the resources [25]. In [26], an knapsack approach is presented for the allocation of virtual machines in the sense to improve the energy efficiency.

All the methods discussed above have the problem in achieving higher performance in scheduling, energy efficiency and time complexity.

Real Time Multi Factor Energy Approximation based Scheduling

The proposed multi factor energy approximation based scheduling algorithm maintains the traces of different tasks being performed by the resources in earlier time. According to the traces available, the method classifies the resources based on the value of energy consumption, task support and so on. With the given set of task, for each task, the method identifies the virtual machine based on the resource available. For any resource available in the cloud, the method computes different task support measures (TSM) for various resources like CPU, Memory and Bandwidth. Also, for the submitted task, the method computes the task requirement support towards various cloud resources. Using the task support measure, the method performs energy approximation for the resources. Using both the measures, the method performs scheduling of resources according to the requirement support value.

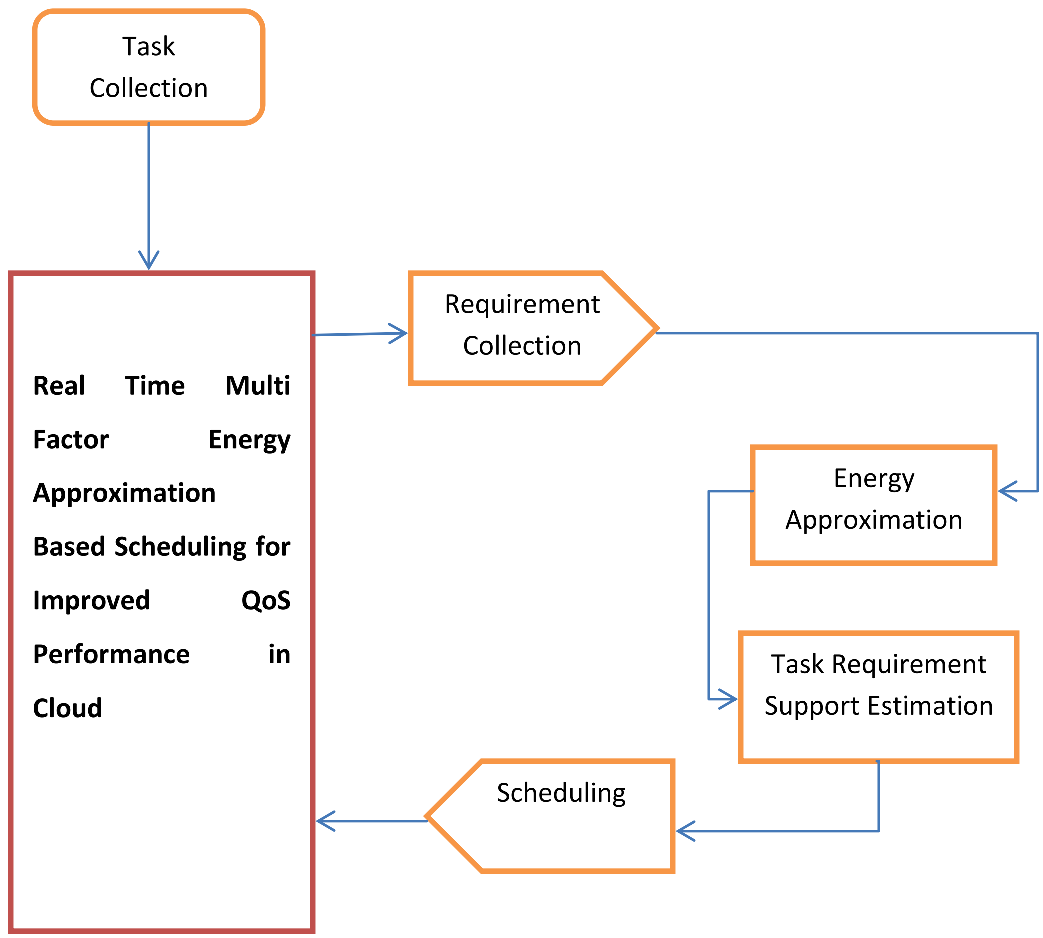

Figure 1: Architecture of proposed Real Time Multi Factor Energy Approximation Based Scheduling

The functional architecture of proposed RTMFEA scheduling is presented in Figure 1, and the functional components are detailed in this section.

Requirement Collection

The task set provided by the user has been received by the model which first identifies the list of task submitted. Second, for each task, the required resources are identified. According to that, the list of all virtual machines

DOI: 10.5373/JARDCS/V12I8/20202466
ISSN 1943-023X
Received: 16 June 2020/Accepted: 18 July 2020

available and the underlying resources present in the VM are identified. Such list of instances and virtual machines are approximated for their energy in the next stages.

Consider, the task set Ts is submitted by the user where each task T_i requires a set of resources of class from the set Cs {C1, C2, C3) where the classes represents the type of resource like CPU, Memory and Bandwidth. From the resource set Cs, the task T_i would require a subset of resources to complete the task. Any task T_i would contain number of instructions to be executed over the resources mentioned and identified by this stage. There will be N number of resource set would be available which can be named as Vms $\{Vm_1, Vm_2, \ldots, Vm_n\}$ where each virtual machine would be having different resources of class Cs. However, it is necessary to identify the optimal resource or virtual machine to complete the task identified. According to the model, the list of resources and their instances are identified to perform scheduling further.

Energy Approximation

The energy is the dominant factor to be considered in accessing the cloud resource. Any resource being accessed by the user would claim certain amount of energy to execute the task. But, not all the resources claims same set of energy and vary accordingly. In order to improve the QoS performance, the energy claimed by the resources should be considered. In this case, the method first identifies the set of virtual machines Vms and for each resource under each instance of Vms, the method identifies the list of traces generated because of previous execution. According to the trace, the method computes the energy claimed by the resource and estimates the task support measure (TSM). According to the energy claim factor (ECF), the resources are sorted based on the value of ECF. Estimated ECF values are given to the scheduler to perform task scheduling.

Pseudo Code of Energy Approximation:
Given: Resource Traces RT, Vm set
Obtain: ECF set.
Start

Read RT, Vms.

For each Vm

Identify the resources under Vm as Rvm = $\sum\{CPU, Memory, Bandwidth\} \in Vm$ (1)

For each resource Ri

Identify the traces from RT as RiT = $\sum_{i=1}^{size(RT)} RT(i).Resource == Ri$ (2)

Compute Task Support Measure TSM_{Ri}.

$TSM_{Ri} = \sum_{i=1}^{size(RiT)} (RiT(i).State == F) \Big/ size(RiT)$ (3)

Compute energy utilized Eu = $\sum_{i=1}^{size(RiT)} RiT(i).Energy \Big/ size(RiT)$ (4)

End

Compute Energy Claim Factor ECF = $\sum Eu \Big/ \sum TSM$ (5)

End

Stop

The energy approximations on different resources under various virtual machines are performed in the above pseudo code. The method identifies the virtual machines and their resources. For each of them the value of task support measure (TSM) and energy utilization is measured. Using both of them, the method computes the method computes the value of energy claim factor. Estimated ECF values are added to the ECS set towards task scheduling.

Task Requirement Support Estimation

The task requirement support is the value which represents the efficiency of any resource on supporting the completion of the task given. To measure the value of TRS, the method first computes the energy claim factor which is measured based on the joules of energy claimed by the resource to execute specific number of statements and the number of data servers to be accessed, number of resources needed and the total make span time. Using all these, the method computes the value of task requirement support for various virtual machines. Estimated values are given to the scheduler towards task scheduling.

DOI: 10.5373/JARDCS/V12I8/20202466
ISSN 1943-023X

Received: 16 June 2020/Accepted: 18 July 2020

Pseudo Code of TRS Estimation

Given: Virtual machine Vm, Resource Trace RT, Task T
Obtain: TRS
Start
Read Vm, RT, T.
Machine Trace Mt = $\sum_{i=1}^{size(RT)} RT(i).MID == Vm$ -- (6)
Compute Energy Claim Factor ECF.
Identify number of statements to be executed Nse = $\sum Statements \in T$ -- (7)
Identify number of data servers required Nds = $\sum Dataservers \tilde{} T$ -- (8)
Compute average statements executed = $\frac{\sum_{i=1}^{size(RT)} Statements \in RT(i).Task}{size(RT)}$ -- (9)
Compute task requirement support TRS = $\frac{NSE}{ECF} \times NDS$ -- (10)
Stop

The task requirement support measure on any task submitted has been measured is presented in the above pseudo code. The method computes the energy claimed by the resource in estimating the number of statements to compute the value of task requirement support. Estimated TRS value has been used to perform scheduling.

RTMFEA Scheduling

The real time multi feature energy approximation scheduling scheme monitors the performance of various resources under the environment. When a task set has been submitted, the method first performs requirement collection and according to that, the method identifies the list of all resources. Further, with the resources identified, the method computes the task requirement support (TRS) for each task submitted and computes the Energy claim factor. Using both of them, the method computes the value of Scheduling Support (SS) based on which a single resource has been selected for any task submitted.

Pseudo Code of RTMFEA Scheduling

Given: Task Set Ts, Resource Trace RT
Obtain: Null
Start
Read RT, Ts.
For each task Ti
Perform requirement collection.
For each resource Ri
ECF = Perform energy approximation.
TRS = Estimate Task requirement support.
Compute SS = ECF×TRS
End
Choose the resource with maximum SS.
End
Stop

The working of RTMFEA scheduling is presented in the above pseudo code which identifies the list of resources required and available. Based on that the method performs energy approximation and estimates the task requirement support values to select a single resource over which the task can be scheduled.

III. Results and Discussion

The proposed real time multi feature energy approximation based scheduling algorithm has been implemented and evaluated for its performance under different simulation conditions. Various scenario of resources and instances has been considered for the evaluation of proposed algorithm. Obtained evaluation results are presented in detail in this section.

Table 1: Evaluation Details

Parameter	Value
Tool Used	Cloud Sim
Number of Resources	20
Number of instances	5
Number of tasks	100

DOI: 10.5373/JARDCS/V12I8/20202466
ISSN 1943-023X

Received: 16 June 2020/Accepted: 18 July 2020

The environmental conditions being used for the evaluation of proposed method has been presented in Table 1. The methods are analyzed for their performance under various parameters and the results obtained are presented in this section.

Table 2: Performance in scheduling

Scheduling Performance % vs No of Tasks			
	25 Tasks	50 Tasks	100 Tasks
LBACO	74	69	63
RNPSO	78	72	68
LIF	84	78	72
RTMFEA	89	93	96

The performance of different methods in scheduling different number of tasks are measured and presented in Table 2. The proposed RTMFEA algorithm has produced higher scheduling performance than other methods.

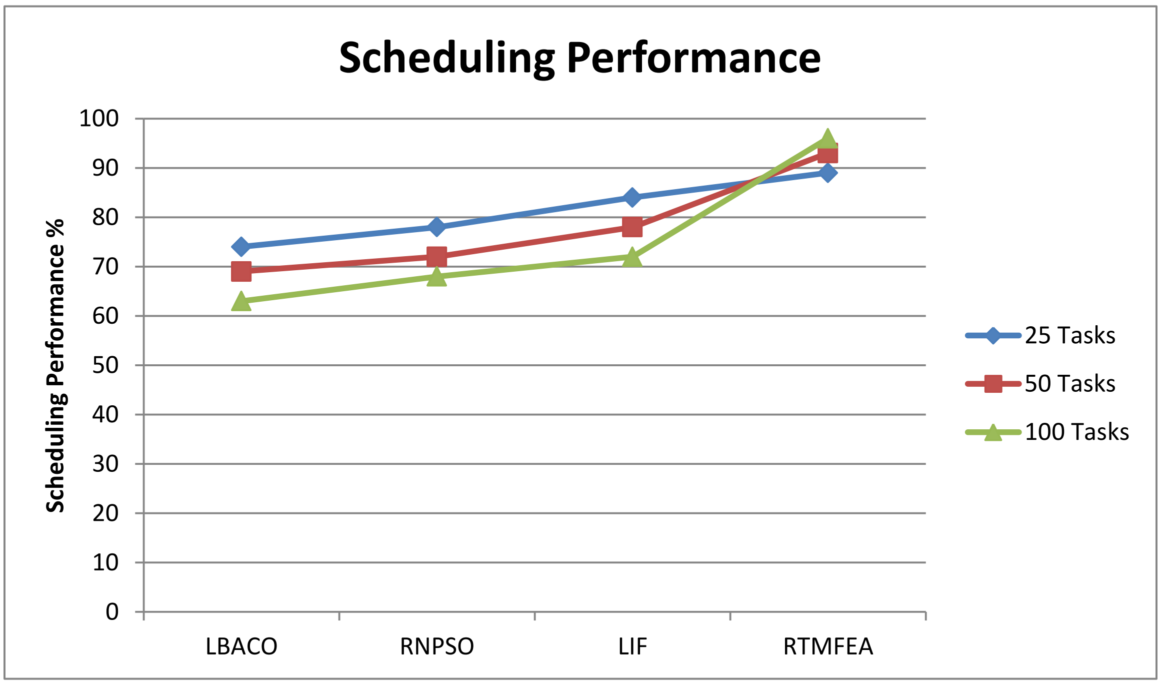

Figure 2: Analysis on Scheduling Performance

The performance in scheduling introduced by different methods on varying number of task submitted are measured and compared in Figure 2. The proposed RTMFEA algorithm has produced higher scheduling performance than other methods.

Table 3: Performance in Energy Efficiency

Energy Efficiency % vs No of Tasks			
	25 Tasks	50 Tasks	100 Tasks
LBACO	72	67	64
RNPSO	76	70	66
LIF	82	76	73
RTMFEA	88	92	95

The performance of different methods in energy efficiency are measured at varying number of tasks and presented in Table 2. The proposed RTMFEA algorithm has produced higher energy efficiency than other methods.

DOI: 10.5373/JARDCS/V12I8/20202466
ISSN 1943-023X

Received: 16 June 2020/Accepted: 18 July 2020

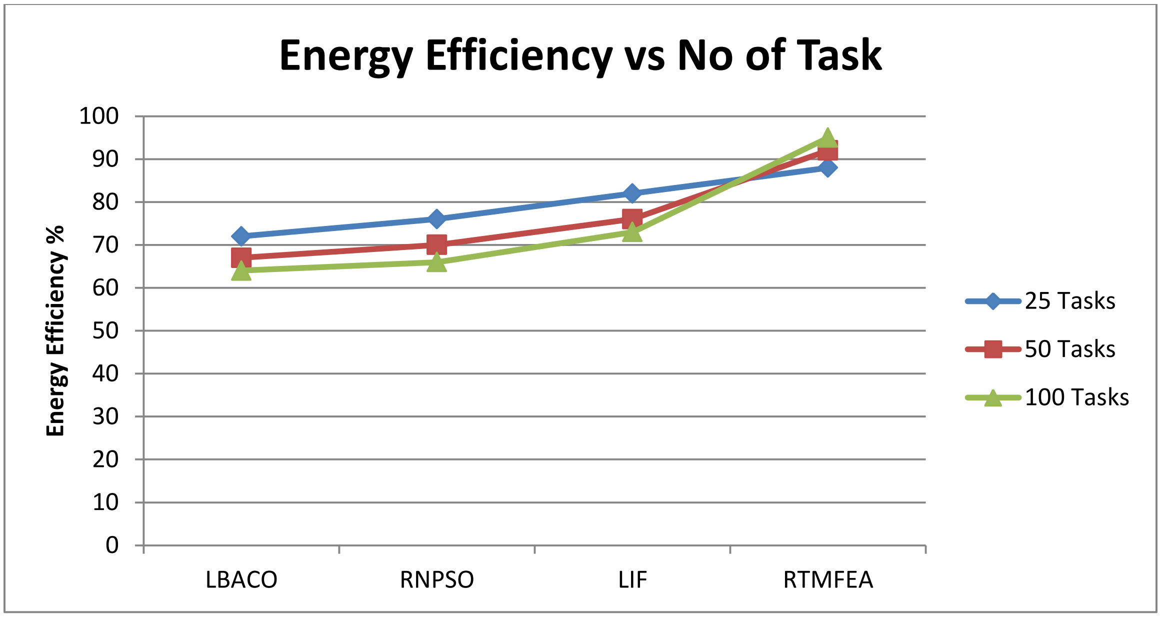

Figure 3: Analysis on Energy Efficiency

The efficiency of the methods on energy has been measured by varying the number of tasks. In each case, the proposed RTMFEA algorithm has produced higher performance than other methods.

Table 4: Performance in Resource Utilization

Resource Utilization Performance vs No of Task			
	25 Tasks	50 Tasks	100 Tasks
LBACO	68	72	76
RNPSO	72	76	79
LIF	77	82	86
RTMFEA	84	91	97

The performance of different methods in utilizing the available resources are measured and presented in Table 4. The proposed RTMFEA algorithm has produced higher utilization performance than other methods at all the conditions.

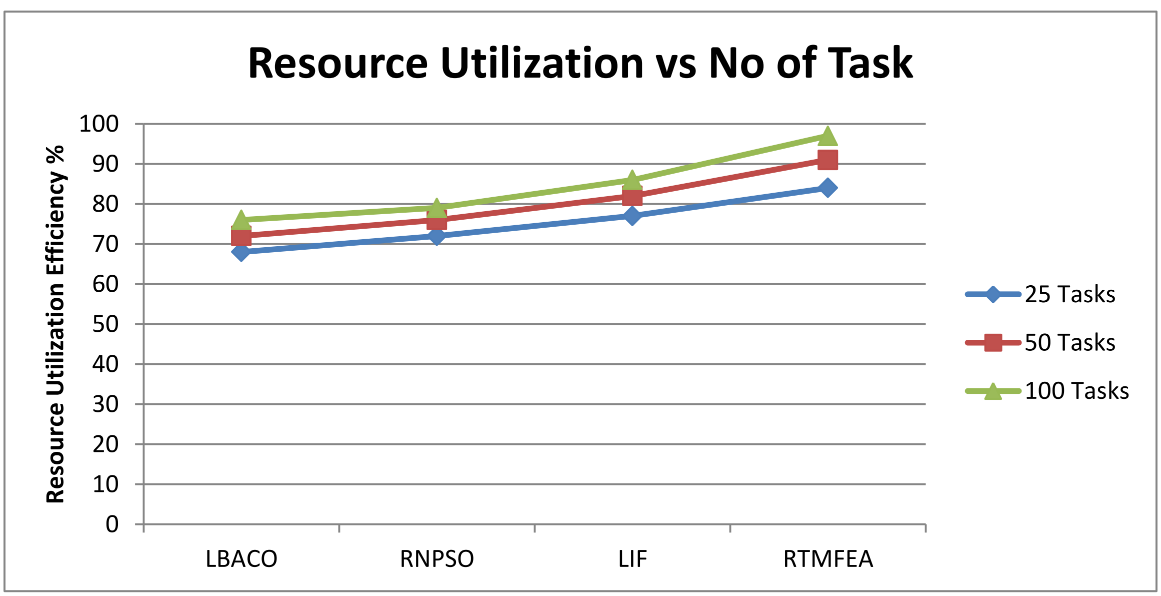

Figure 4: Analysis on Resource Utilization

DOI: 10.5373/JARDCS/V12I8/20202466
ISSN 1943-023X

Received: 16 June 2020/Accepted: 18 July 2020

The performance in resource utilization has been measured and presented in Figure 4, where the proposed RTMFEA algorithm has produced higher resource utilization performance than other methods.

Table 5: Performance in Time Complexity

Time Complexity vs No of Task			
	25 Tasks	50 Tasks	100 Tasks
LBACO	38	52	68
RNPSO	32	46	59
LIF	28	34	46
RTMFEA	24	28	32

The time complexity in scheduling the tasks under available resources are measured and presented in Table 5. The proposed RTMFEA algorithm has produced less time complexity compare to other methods.

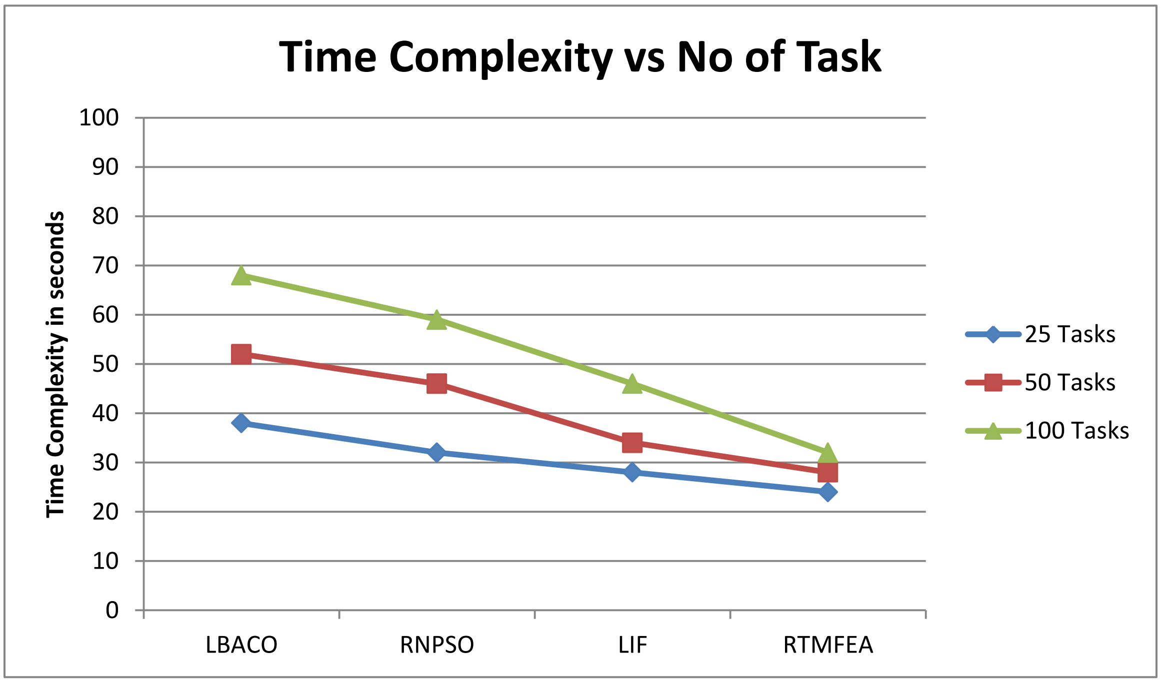

Figure 5: Analysis on Time Complexity

The time complexity introduced by different methods on scheduling the task given has been measured and presented in Figure 5. The proposed RTMFEA algorithm has produced less time complexity than other methods.

IV. Conclusion

This paper presented a real time multi feature energy approximation based scheduling algorithm. First, the method receives the task set and identifies the list of resources required. Also, in each resource available, the type of resources and instances available are identified. Further, for each resource available, the method performs energy approximation and for each task, the method computes the task requirement support values. Using both of them, the method computes the value of scheduling strength based on which the scheduling is performed. The proposed RTMFEA algorithm has produced higher performance in scheduling, energy and resource utilization with less time complexity.

References

[1] Zhang, Q., Cheng, L. and Boutaba, R. Cloud computing: state-of-the-art and research challenges. *Journal of internet services and applications* **1** (1) (2010) 7-18.

[2] Reddy, V.K., Rao, B.T. and Reddy, L.S.S. Research issues in cloud computing. *Global Journal of Computer Science and Technology* **11** (11) (2011).

[3] Gawali, M.B. and Shinde, S.K. Task scheduling and resource allocation in cloud computing using a heuristic approach. *Journal of Cloud Computing* **7** (1): (2018) 1–16.

DOI: 10.5373/JARDCS/V12I8/20202466
ISSN 1943-023X
Received: 16 June 2020/Accepted: 18 July 2020

[4] Tian, W., Liu, X., Jin, C. and Zhong, Y. LIF: A dynamic scheduling algorithm for Cloud data centers considering multi-dimensional resources. *Journal of Information & Computational Science* **10** (12) (2013) 3925-3937.

[5] Chien, C.H., Chang, P.H. and Soo, V.W. Market-oriented multiple resource scheduling in grid computing environments. *In IEEE 19th International Conference on Advanced Information Networking and Applications (AINA'05),* 2005, 867-872.

[6] Fujiwara, I., Aida, K. and Ono, I. Applying double-sided combinational auctions to resource allocation in cloud computing. *In 10th IEEE/IPSJ International Symposium on Applications and the Internet*, 2010, 7-14.

[7] Yang, Z., Yin, C. and Liu, Y. A cost-based resource scheduling paradigm in cloud computing. *In IEEE 12th International Conference on Parallel and Distributed Computing, Applications and Technologies*, 2011, 417-422.

[8] Ellendula M. and Thirumalaisamy R. Efficient Scheduling Algorithm for Cloud. *Big Data Cloud and Computing Challenges Procedia Computer Science* **50** (2015) 353-356.

[9] Zhang, R., Wu, K., Li, M. and Wang, J. Online resource scheduling under concave pricing for cloud computing. *IEEE Transactions on Parallel and Distributed Systems* **27** (4) (2015) 1131-1145.

[10] Tawfeek, M.A., El-Sisi, A., Keshk, A.E. and Torkey, F.A. Cloud task scheduling based on ant colony optimization. *In IEEE 8th international conference on computer engineering & systems (ICCES)*, 2013, 64-69.

[11] Gondhi, N.K. and Sharma, A. Local search based ant colony optimization for scheduling in cloud computing. *In IEEE Second International Conference on Advances in Computing and Communication Engineering*, 2015, 432-436.

[12] Li, H.H., Fu, Y.W., Zhan, Z.H. and Li, J.J. Renumber strategy enhanced particle swarm optimization for cloud computing resource scheduling. *In IEEE Congress on Evolutionary Computation (CEC)*, 2015, 870-876.

[13] Rodriguez, M.A. and Buyya, R. Deadline based resource provisioning and scheduling algorithm for scientific workflows on clouds. *IEEE transactions on cloud computing* **2** (2) (2014) 222-235.

[14] Sahni, J. and Vidyarthi, D.P. A cost-effective deadline-constrained dynamic scheduling algorithm for scientific workflows in a cloud environment. *IEEE Transactions on Cloud Computing* **6** (1) (2015) 2-18.

[15] Potluri, S. and Rao, K.S. Quality of service based task scheduling algorithms in cloud computing. *International Journal of Electrical and Computer Engineering* **7** (2) (2017) 1088-1095.

[16] Ma, J., Li, W., Fu, T., Yan, L. and Hu, G. A novel dynamic task scheduling algorithm based on improved genetic algorithm in cloud computing. *In Wireless Communications, Networking and Applications, Springer, New Delhi*, 2016, 829-835.

[17] Thomas, A., Krishnalal, G. and Raj, V.J. Credit based scheduling algorithm in cloud computing environment. *Procedia Computer Science* **46** (2015) 913-920.

[18] Zhao, T. and Jing, M. Bandwidth-aware multi round task scheduling algorithm for cloud computing. *Journal of Intelligent & Fuzzy Systems* **31** (2) (2016) 1053-1063.

[19] Kaur, T. and Chana, I. Green Sched: An intelligent energy aware scheduling for deadline-and-budget constrained cloud tasks. *Simulation Modelling Practice and Theory* **82** (2018) 55-83.

[20] Meena, J., Kumar, M. and Vardhan, M. Cost effective genetic algorithm for workflow scheduling in cloud under deadline constraint. *IEEE Access* **4** (2016) 5065-5082.

[21] Ebadifard F. and Babamir S.M. A PSO-based task scheduling algorithm improved using a load-balancing technique for the cloud computing environment. *Concurr. Comput.*, 2017, 1-16.

[22] Chen, Z.G., Du, K.J., Zhan, Z.H. and Zhang, J. Deadline constrained cloud computing resources scheduling for cost optimization based on dynamic objective genetic algorithm. *In IEEE Congress on Evolutionary Computation (CEC),* 2015, 708-714.

[23] Praveenchandar, J. and Tamilarasi, A. Dynamic resource allocation with optimized task scheduling and improved power management in cloud computing. *Journal of Ambient Intelligence and Humanized Computing*, 2020, 1-13.

[24] Chou, L.D., Chen, H.F., Tseng, F.H., Chao, H.C. and Chang, Y.J. DPRA: dynamic power-saving resource allocation for cloud data center using particle swarm optimization. *IEEE Systems Journal* **12** (2) (2016) 1554-1565.

[25] Dabbagh, M., Hamdaoui, B., Guizani, M. and Rayes, A. Energy-efficient resource allocation and provisioning framework for cloud data centers. *IEEE Transactions on Network and Service Management* **12** (3) (2015) 377-391.

[26] Han, S., Min, S. and Lee, H. Energy efficient VM scheduling for big data processing in cloud computing environments. *Journal of Ambient Intelligence and Humanized Computing*, 2019, 1-10.

DOI: 10.5373/JARDCS/V12I8/20202466
ISSN 1943-023X

Received: 16 June 2020/Accepted: 18 July 2020