QoS Analysis with Support over Energy Efficient and Cost based Workflow Scheduling Algorithms in Multi Cloud Environment

Premnadh Katkam

Research Scholar
Department of Computer Science and Engineering,
Annamalai University,
Annamalainagar - 608002.

EMAIL: PREMS.1216@GMAIL.COM

Dr.P.Anbalagan

Assistant Professor
Dept. of Computer Science & Engineering,
Annamalai University
Annamalainagar – 608 002.
EMAIL: ANBALAGANSAMY@GMAIL.COM

Dr.V.V.S.S.S.Balaram

Professor & Head
Sreenidhi Institute of Science and Technology
Yamnampet, Ghatkesar,
HYDERABAD – 501301.
EMAIL: VBALARAM@SREENIDHI.EDU.IN

Article Info
Page Number: 407-422
Publication Issue:
Vol. 71 No. 4 (2022)

Article History
Article Received: 25 March 2022
Revised: 30 April 2022
Accepted: 15 June 2022
Publication: 19 August 2022

Abstract

Cloud computing provides tremendous infrastructure facility for the execution of scientific workflows and commercial applications by offering dynamic scalable, reliable and flexible computing platform. Almost all the scientific application is modeled as workflows. Dynamic scheduling workflow applications with the optimal cost and execution time driven environment is a challenging task. The important point to be considered for the execution of performance driven applications requires suitable scheduling algorithms with optimal resources with minimum execution time with specified Quality of Service (QoS). In this proposed work, a Cost Based Workflow Scheduling Algorithm (CBWS) and Energy Efficient Workflow Scheduling Algorithm (EEWS) are designed. The simulation results confirm that the proposed scheduling algorithm outperforms the related well-known scheduling approaches. Further, the performance comparison demonstrates its performance superiority over the existing methods.

Keywords: Energy Efficiency, Cost Analysis, Scheduling Approaches, QoS

1.0 INTRODUCTION

Cloud computing is a large-scale distributed computing driven by economies, in which the resources and services are delivered on demand to external customers via the Internet. Cloud computing has emerged as an effective and efficient way of resource provisioning. Due to the centralized management of infrastructures in cloud platforms, users can access on-demand resources, charged in a “pay-as-you-go” manner.

Consumers and operators are generating large amount of data on various services per minute in cloud environment, which increasingly comes to show all typical properties of big data. The technological advantages of clouds, such as elasticity, scalability, accessibility and reliability, have made them an attractive alternative to replace private in-house IT infrastructures. It has emerged as promising computing paradigm for government, research institute and industry to solve ever-increasing computing and storage problems.

Cloud can usually provide three kinds of services such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). These services are offered with different service levels so as to meet the needs of various customer groups. Although many cloud services have a similar functionality (e.g., computing services, storage services, network services, etc.), they differ from each other by non-functional qualities termed QoS parameters, such as service time, service cost, service availability, service energy consumption, service utilization, and so forth. These QoS parameters may be defined and proposed by different Service Level Agreements (SLA). Cloud computing presents an interesting technology that facilitates the execution of scientific and commercial applications.

A large number of scientific applications can be modeled by workflows in many domains, including bioinformatics, astronomy, astrophysics, and high-energy physics. Such scientific workflows can benefit from large-scale cloud infrastructures. Workflow is defined as a collection of tasks that are processed in a specific order to accomplish a real application, and scientific workflow has been proved as an efficient and popular method to model various scientific computing problems in parallel and distributed systems.

A single scientific workflow usually contains hundreds or thousands of tasks, thereby requiring a large amount of computing resources for execution. Fortunately, those resources can be provisioned by the cloud infrastructures. However, the tasks contained in scientific workflows have dependencies and communications among them, differing significantly from the unrelated tasks.

These scientific applications contain a great number of tasks that have precedence constraints. They can be defined by Directed Acyclic Graph (DAG). These scientific workflows typically involve complex data of different sizes and long term computer simulations. They need high computation power and the availability of large infrastructures that grid and more recently cloud computing environments provide with different QoS levels. Due to the importance of workflow applications, several research projects have been conducted to develop workflow management systems with scheduling algorithms.

The cloud management system needs to allocate resource for scientific workflow executions and multi-cloud environment can be viewed as a type of platform service facilitating the automation of scientific and commercial applications on the cloud by masking their orchestrations and executions. In order to effectively schedule the tasks and data applications on these cloud environments, workflow management systems require more elaborated scheduling strategies to meet QoS constraints and the precedence relationships between workflow tasks.

Therefore, within a cloud platform, the computing resources are provided in the form of Virtual Machines (VMs). The VMs are usually provided in various specifications, which are measured by several configuration parameters including the number of CPU cores, the amount of memory, the disk capacity, etc.

1.1 OBJECTIVE OF COST AND ENERGY EFFICIENCY APPROACH

The study of workflow scheduling is becoming an important challenge in the area of cloud computing. The workflow scheduling in the cloud is a difficult problem. This problem is even more difficult when there are several factors to be considered namely, (1) the various QoS requirements of customers like service response time, service cost, and so forth; (2) the heterogeneity, dynamicity and elasticity of cloud services; (3) the various ways of combining these services to execute workflow tasks; (4) the transfer of large volumes of data. However, the workflow scheduling problem is seen as a combinatorial problem, where it is impossible to find the globally optimal solution by using simple algorithms or rules.

The main objective of this paper is to design three different workflow scheduling algorithms to optimize the important parameters such as makespan, cost and energy. Most of the previous works have concentrated only on two QoS parameters namely, the deadline and budget. In this paper, the work is extended to handle multiple QoS requirements. In this proposed research, the QoS-based workflow scheduling is addressed which aims to minimize the cost and total time execution of user applications as specified in the SLA. Furthermore, the scheduler must also be able to schedule workflow tasks so as to maximize the provider profits by minimizing energy consumption while preserving the user's QoS preferences.

2.0 RELATED WORK

Estimation of the execution time is an important part of the workflow scheduling problem. The aim of this work [1] is to highlight common problems in estimating the workflow execution time and propose a solution that takes into account, the complexity and the stochastic aspects of the workflow components as well as their runtime. The solution proposed in this work addresses the problems at different levels from a task to a workflow, including the error measurement and the theory behind the estimation algorithm. The proposed makespan estimation algorithm can be integrated easily into a wide class of schedulers as a separate module. A dual stochastic representation, characteristic/distribution function, in order to combine task estimates into the overall workflow makespan used. Additionally, the workflow reductions-operations on a workflow graph that do not decrease the accuracy of the estimates but simplify the graph structure, hence increasing the performance of the algorithm are proposed. Another very important feature of this work is that the described estimation schema is integrated into earlier developed scheduling algorithm GAHEFT and experimentally evaluates the performance of the enhanced solution in the real environment using the CLAVIRE platform.

The Max-Min algorithm is built based on comprehensive study of the impact of RASA algorithm in scheduling tasks and the atom concept of Max-min strategy [2]. An Improved version of Max-min algorithm is proposed to outperform scheduling map at least similar to RASA map in total complete time for submitted jobs. Improved Max-min is based on the expected execution time instead of complete time as a selection basis. Experimental results show availability of load balance in small cloud computing environment and total small makespan in large-scale distributed system like cloud computing. In turn, scheduling tasks within cloud computing using Improved Max-min demonstrates achieving schedules with comparable lower makespan rather than RASA and original Max-min.

Workflow scheduling is recognized as well known NP-complete problem in the perspective of cloud computing environment. Workflow applications always need high compute-intensive operations because of the presence of precedence-constrains. The scheduling objective is to map the workflow application to the VMs pool at available cloud datacenters such that the overall processing time (makespan) is to be minimized and average cloud utilization is maximized. In this work [3], a two phase workflow scheduling algorithm with a new priority scheme is proposed. It considers the ratio of average communication cost to the average computation cost of the task node as a part of prioritization process in the first phase. Prioritized tasks are mapped to suitable virtual machines in the second phase. Proposed algorithm is capable of scheduling large size workflows in heterogeneous multi-cloud environment. The proposed algorithm is simulated rigorously on standard scientific workflows and simulated results are compared with the existing dependent task scheduling algorithms as per the assumed cloud model. The results remarkably show that the proposed algorithm supersedes the existing algorithms in terms of makespan, speed-up, schedule length ratio and average cloud utilization.

Large-scale applications expressed as scientific workflows are often grouped into ensembles of interrelated workflows. In this work [4], a new and important problem concerning the efficient management of such ensembles is addressed under budget and deadline constraints on IaaS clouds. IaaS clouds are characterized by on-demand resource provisioning capabilities and pay-per-use model. The novel algorithms based on static and dynamic strategies for both task scheduling and resource provisioning is developed and discussed. The performance evaluation via simulation using a set of scientific workflow ensembles with a broad range of budget and deadline parameters, taking into account task granularity, uncertainties in task runtime estimations, provisioning delays, and failures. The key factor determining the performance of an algorithm is its ability to decide which workflows in an ensemble to admit or reject for execution. The results show that an admission procedure based on workflow structure and estimates of task runtimes can significantly improve the quality of solutions.

The emergence of Cloud Computing as a model of service provisioning in distributed systems instigated researchers to explore its pros and cons on executing different large scale scientific applications, i.e., Workflows. One of the most challenging problems in cloud is to execute workflows while minimizing the execution time as well as cost incurred by using a set of heterogeneous resources over the cloud simultaneously. In this work [5], Budget and

Deadline Constrained Heuristic based upon Heterogeneous Earliest Finish Time (HEFT) is presented to schedule workflow tasks over the available cloud resources. The proposed heuristic presents a beneficial trade-off between execution time and execution cost under given constraints. The proposed heuristic is evaluated for different synthetic workflow applications by a simulation process and comparison is done with state-of-art algorithm i.e., HEFT. The simulation results show that the proposed scheduling heuristic can significantly decrease the execution cost while producing makespan as good as the best known scheduling heuristic under the same deadline and budget constraints.

Effective scheduling is one of the key concerns while executing workflows in the cloud environment. Workflow scheduling in clouds refers to the mapping of workflow tasks to the cloud resources to optimize some objective function. In this work [6], a recently developed meta-heuristic method called the BAT algorithm is applied to solve the multi-objective problem of workflow scheduling in clouds that minimizes the execution time and maximizes the reliability by keeping the budget within user specified limit. Comparison of the results is made with basic, randomized, evolutionary algorithm (BREA) that uses greedy approach to allocate resources to the workflow tasks on the basis of low cost, high reliability and improved execution time machines. It is clear from the experimental results that the BAT algorithm performs better than the basic randomized evolutionary algorithm.

Scientific workflows are often deployed across multiple cloud computing platforms due to their large-scale characteristics. This can be technically achieved by expanding a cloud platform. However, it is still a challenge to conduct scientific workflow executions in an energy-aware fashion across cloud platforms or even inside a cloud platform, since the cloud platform expansion will make the energy consumption a big concern. In this work [7], an Energy-aware Resource Allocation method, named EnReal, is proposed to address the above challenge. Basically, the dynamic deployment of virtual machines for scientific workflow executions is leveraged. Specifically, an energy consumption model is presented for applications deployed across cloud computing platforms, and a corresponding energy-aware resource allocation algorithm is proposed for virtual machine scheduling to accomplish scientific workflow executions. Experimental evaluation demonstrates that the proposed method is both effective and efficient.

Big data applications require a suitable platform like Cloud to execute the deadline-constrained scientific workflows which are typical and often require many hours to finish. Moreover, the problem of energy consumption is one of the major concerns in clouds. In this work [9], a Cost and Energy Aware Scheduling (CEAS) algorithm is presented for cloud scheduler to minimize the execution cost of workflow and reduce the energy consumption while meeting the deadline constraint. The CEAS algorithm consists of five sub-algorithms. First, the VM selection algorithm is used which applies the concept of cost utility to map tasks to their optimal virtual machine (VM) types by the sub-makespan constraint. Then, two tasks merging methods are employed to reduce execution cost and energy consumption of workflow. Further, in order to reuse the idle VM instances which have been leased, the VM reuse policy is also proposed. Finally, the scheme of slack time reclamation is utilized to save energy of leased VM instances. According to the time complexity analysis, it is concluded that the time complexity of each sub-algorithm is polynomial. The CEAS algorithm is evaluated using Cloudsim for four real-world scientific workflow applications, which demonstrates that it outperforms the related well-known approaches.

The growth of energy consumption has been explosive in current data centers, super computers, and public cloud systems. This explosion has led to greater advocacy of green computing, and many efforts and works focus on the task scheduling in order to reduce energy dissipation. In order to obtain more energy reduction as well as maintain the quality of service by meeting the deadlines, this work [10] proposes a DVFS-enabled Energy-efficient Workflow Task Scheduling algorithm (DEWTS). Through merging the relatively inefficient processors by reclaiming the slack time, DEWTS can leverage the useful slack time recurrently after severs are merged. DEWTS firstly calculates the initial scheduling order of all tasks, and obtains the whole makespan and deadline based on Heterogeneous-Earliest-Finish-Time (HEFT) algorithm. Through resorting the processors with their running task number and energy utilization, the underutilized processors can be merged by closing the last node and redistributing the assigned tasks on it. Finally, in the task slacking phase, the tasks can be distributed in the idle slots under a lower voltage and frequency using DVFS technique, without violating the dependency constraints and increasing the slacked makespan. Based on the amount of randomly generated DAGs workflows, the experimental results show that DEWTS can reduce the total power consumption by up to 46.5 % for various parallel applications as well as balance the scheduling performance.

Cloud computing provides the users with enormous heterogeneous computing resources for the execution of large scale applications. It is vital to determine the viability of the proposed algorithms on high end virtualized infrastructure. However, it is complex to perform repeated evaluations on real cloud platform, to analyze the performance of proposed algorithms. Simulations are the widely accepted way for conducting experiments to assess the performance. WorkflowSim toolkit is a popular simulation framework that has been selected as a cloud simulation platform. The simulation is performed for two real world scientific workflow applications such as Montage and Cybershake.

3.0 DESIGN OF QoS BASED SCHEDULING ALGORITHM

Scientific workflows are generally used for representing complex distributed scientific computations. They are composed of a large number of tasks and the execution of these tasks may require many complex modules and software. The performance evaluation of workflow optimization techniques in real infrastructures is complex and time consuming. As a result, simulation-based studies have become one of the most popular methods to evaluate the scientific workflows. This approach reduces the complexity of the experimental setup and saves much effort in workflow execution by enabling the testing of their applications in a repeatable and controlled environment. Existing workflow simulators fail to provide a framework that takes heterogeneous system overheads and failures into consideration. They also lack the support for widely used workflow optimization techniques such as task clustering. WorkflowSim extends the existing CloudSim simulator by providing a higher layer of workflow management. CloudSim only supports the execution of single workloads while WorkflowSim focuses on workflow scheduling and execution. CloudSim has a simple model of task execution that does not consider task dependencies or clustering. It also ignores the occurrence of failures and overheads. WorkflowSim extends CloudSim to fulfill these new requirements (Table 1).

Table 1 QoS Parameters for Scientific Workflow Scheduling

QoS Parameters	**Description**
Makespan[12]	The period between the starting time of the execution and the completion time of the actual workflow.
Price[6]	The amount paid by users for executing workload on cloud providers' service
Throughput[10]	Total number of user request completed within a specified time limit
Resource Utilization[3]	Appropriate use of resources
Turnaround Time[5]	The difference between the completion time and the task submission time
Resource Availability[13]	Resource availability for mapping tasks
Load balancing[8]	Optimization of resources used to reduce the load on the cloud resources
Response time[6]	Time difference between the task arrival and task submission
Energy[5] consumption	Energy utilization during the scheduling processes

The primary challenge in workflow scheduling is to monitor and maintain these QoS parameters. The review of existing research scheduling algorithms as well as enhanced energy and cost effective parameter based workflow scheduling algorithm schemes that consider the QoS parameters are discussed.

4.0 COST AND ENERGY AWARE SCHEDULING ALGORITHM FOR SCIENTIFIC WORKFLOWS WITH DEADLINE CONSTRAINT OVER CLOUD

Big data applications require a suitable platform like Cloud to execute the deadline-constrained scientific workflows which are typical and often require many hours to finish.

4.1 Proposed Cost Based Workflow Scheduling (CBWS)

CBWS scheduler flow diagram is depicted in Fig. 1.

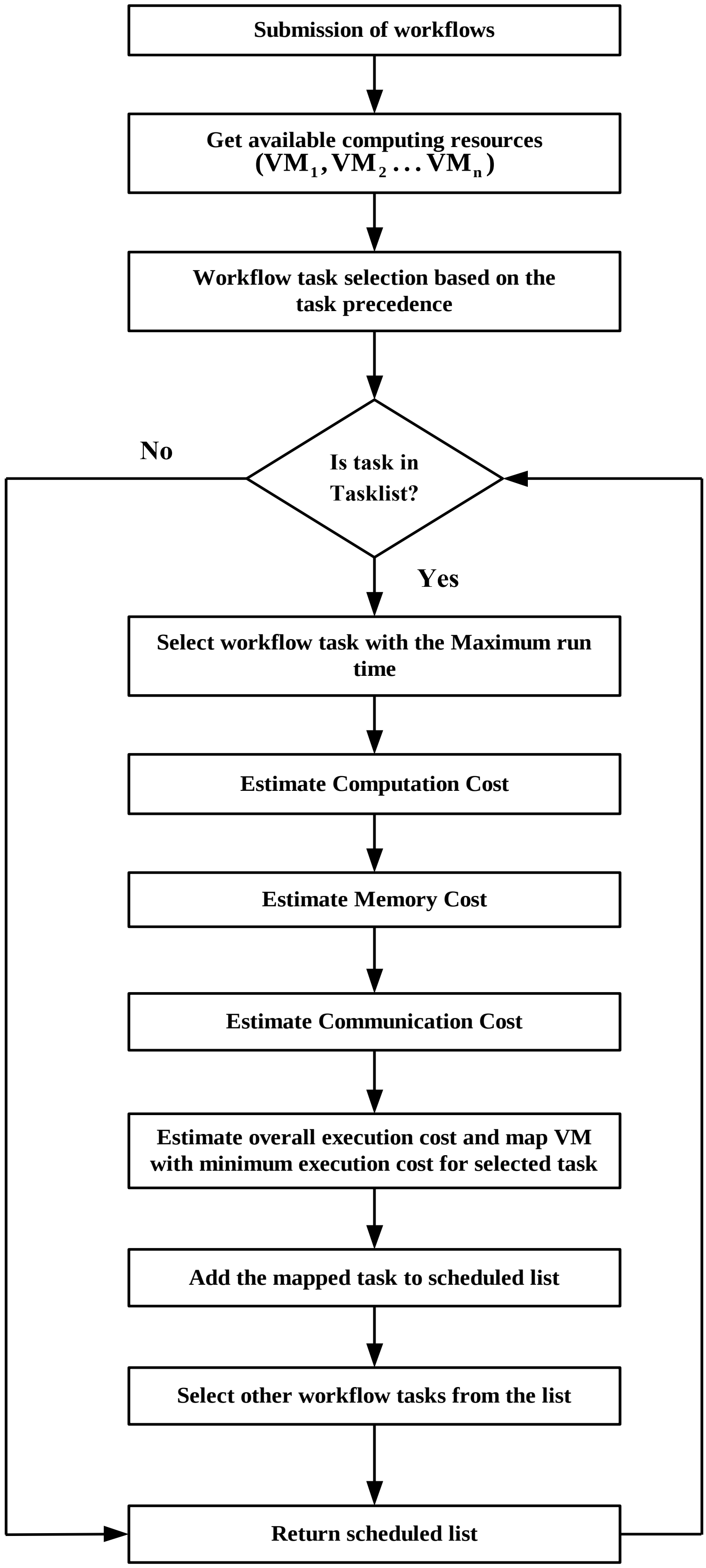

Fig. 1 CBWS Scheduler Flow Diagram

Input : Scientific Workflow Application in DAX

Output: Minimum Cost Execution Schedule

1: Let N be the number of tasks in a workflow to be scheduled

/* Task Selection Phase*/

2: Task list contains N tasks T[N] <- {t_1, t_2, t_3,.....,t_N}

3: For each task t_i from tasklist T (where i<- 0 to N)

4: Select the task with maximum length as t_{max}

5: End For

/* Resource Selection Phase*/

6: Resource List contains K virtual machines R[K]<-{vm_1,vm_2,vm_3,.......,vm_K}

7: For each VM vm_j from resource list R (where j <- 0 to K)

8 : Calculate Expected Execution Time for task $Task_{Max}$ on VM_j as EE_{Time}

9 : EE_{time}<- $l(t_{max})/ \alpha(vm_j)$, where $l(t_i)$ is the length of the task t_{max}, α is the VM Capacity

10 : Calculate Expected Communication Time of task t_{max} to transfer input and output files as EC_{Time}

11 : EC_{Time}<- $data(t_i)$ / $\beta(vm_j)$, where $data(t_i)$ represent the input and output data to be transferred by the task t_i and β is the VM Bandwidth

12 : Calculate the Total Expected Computation Time to compute $Task_{Max}$ on vm_j as TEC_{time}

13 : TEC_{time}<- EE_{Time} + EC_{Time}

14 : Calculate the Expected Computation cost to execute $Task_{Max}$ on vm_j as $Comp_{cost}$

15 : $Comp_{cost}$ = TEC_{time} * VM_{cost}, where VM_{cost} refers to the resource cost based on VM Type

16 : Calculate the Expected Memory cost to execute $Task_{Max}$ on vm_j as Mem_{cost}

17 : Mem_{cost} = TEC_{time} * $data(t_i)$ *costPerMemory, where costPerMemory refers to cost assigned for memory usage

18 : Calculate the Expected Communication cost to execute $Task_{Max}$ on vm_j as $Comm_{cost}$

19: $Comm_{cost}$ = EC_{Time} * BW_{cost} where BW_{cost} refers to cost assigned for bandwidth usage

20: Calculate the Expected Execution cost to execute $Task_{Max}$ on vm_j as Exe_{cost}

21: Exe_{cost} = $Comp_{cost}$+ Mem_{cost}+ $Comm_{cost}$

/* Resource Allocation Phase*/

ISSN: 2094-0343
2326-9865

22: If vm_j status is IDLE && Exe_{cost} is minimum

23: Assign VM_{min}<- vm_j

24: End If

25: Set VM_{min} status as BUSY

26: Allocate VM_{min} to t_{max}

27: End For

Cost Based Workflow Scheduling Algorithm

Cost Based Workflow Scheduling is used to schedule the workflow tasks to the resources based on the estimation of the overall cost that is likely to be incurred for the task execution in the selected resource.

The task with maximum runtime is selected for execution first to map it to the task of high capacity to avail maximum resource utilization. The VM selection is based on the cost of execution on the selected resource. The overall cost of execution depends on the computation cost, memory cost and communication cost. The expected execution cost is calculated for each VM and resource with minimum cost is selected for execution of maximum length task (Fig. 2).

========== OUTPUT ==========

Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start Time	Finish Time	Depth
25	SUCCESS	2	1	0.45	0.1	0.55	0
1	SUCCESS	2	3	4.55	0.55	5.1	1
2	SUCCESS	2	1	5.66	0.55	6.21	1
10	SUCCESS	2	3	3.92	6.21	10.14	2
4	SUCCESS	2	0	14.93	0.55	15.48	1
3	SUCCESS	2	2	17.28	0.55	17.82	1
12	SUCCESS	2	0	5.59	15.48	21.07	2
0	SUCCESS	2	4	23.75	0.55	24.3	1
7	SUCCESS	2	2	13.94	17.82	31.76	2
11	SUCCESS	2	0	12.95	21.07	34.02	2
8	SUCCESS	2	2	2.55	31.76	34.32	2
5	SUCCESS	2	4	18.85	24.3	43.15	2
13	SUCCESS	2	3	25.98	17.82	43.8	2
6	SUCCESS	2	0	13.61	34.02	47.63	2
9	SUCCESS	2	1	43.05	6.21	49.26	2
14	SUCCESS	2	2	0.91	49.26	50.17	3
15	SUCCESS	2	1	1.71	50.17	51.88	4
20	SUCCESS	2	1	3.33	51.88	55.22	5
18	SUCCESS	2	2	13.73	51.88	65.61	5
19	SUCCESS	2	0	15.95	51.88	67.83	5
17	SUCCESS	2	4	18.83	51.88	70.72	5
16	SUCCESS	2	3	25.55	51.88	77.43	5
21	SUCCESS	2	0	13.38	77.43	90.81	6
22	SUCCESS	2	1	4.2	90.81	95.02	7
23	SUCCESS	2	0	37.35	95.02	132.37	8
24	SUCCESS	2	2	0.63	132.37	132.99	9

Total Execution Cost :: 2059.1235073071743

Fig. 2CBWS Scheduler Output

4.2 ENERGY-EFFICIENT SCHEDULING FOR TASKS WITH DEADLINE IN VIRTUALIZED ENVIRONMENTS

Data centers, as resource providers, take advantage of virtualization technology to achieve excellent resource utilization, scalability, and high availability [8]. However, large numbers of computing servers containing virtual machines of data centers consume a tremendous amount of energy. Thus, it is necessary to significantly improve resource utilization. Among the many issues associated with energy, scheduling plays a very important role in successful task execution and energy consumption in virtualized environments. This work seeks to implement an energy-efficient task scheduling algorithm for virtual machines with changeless speed comprised of two main steps such as assigning as many tasks as possible to virtual machines with lower energy consumption and keeping the makespan of each virtual machine within a deadline. A novel scheduling algorithm is proposed for heterogeneous virtual machines in virtualized environments to effectively reduce energy consumption and finish all tasks before a deadline. The new scheduling strategy is simulated using the CloudSim toolkit package. Experimental results show that the proposed approach outperforms previous scheduling methods by a significant margin in terms of energy consumption.

4.3 Proposed Energy Efficient Workflow Scheduling (EEWS)

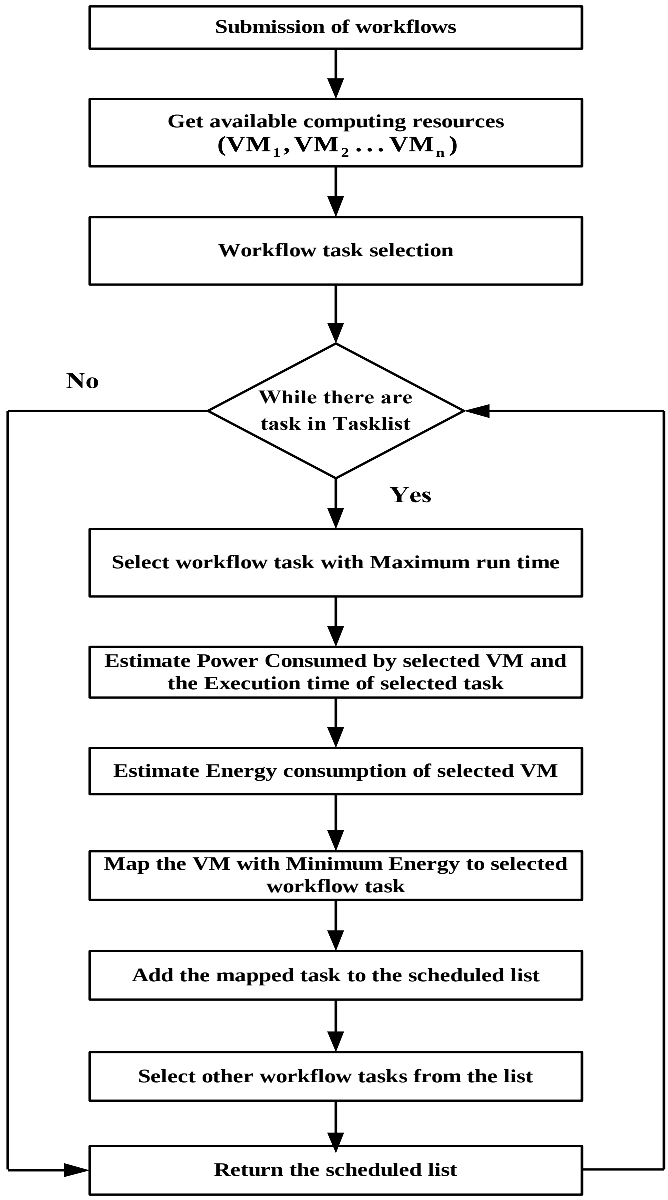

Fig. 3 EEWS Scheduler Flow Diagram

Input : Scientific Workflow Application in DAX
Output: Minimum Makespan Execution Schedule
1: Let N be the number of tasks in a workflow to be scheduled
/* Task Selection Phase*/
2: Task list contains N tasks T[N] <- {t_1, t_2, t_3,.....,t_N}
3: For each task t_i from tasklist T (where i<- 0 to N)
4: Select the task with maximum length as t_{max}
5: End For
/* Resource Selection Phase*/
6: Resource List contains K virtual machines R[K]<-{vm_1,vm_2,vm_3,.......,vm_K}
7: For each VM vm_j from resource list R (where j <- 0 to K)
8 : Calculate Expected Execution Time for task $Task_{Max}$ on VM_j as EE_{Time}
9 : EE_{time}<- $l(t_{max})/ \alpha(vm_j)$, where $l(t_i)$ is the length of the task t_{max}, α is the VM Capacity
10 : Calculate Expected Communication Time of task t_{max} to transfer input and output files as EC_{Time}
11 : EC_{Time}<- $data(t_i)$ / $\beta(vm_j)$, where $data(t_i)$ represent the input and output data to be transferred by the task t_i and β is the VM Bandwidth
12 : Calculate the Total Expected Computation Time to compute $Task_{Max}$ on vm_j as TEC_{time}
13 : TEC_{time}<- $EE_{Time} + EC_{Time}$
14 : Calculate the Expected Power consumed by VM to execute $Task_{Max}$ on vm_j as VM_{Power}
15 : $VM_{Power} = P_{Idle} + (P_{Full} – P_{Idle})* VM_{Util}$
16 : Calculate the Expected Energy consumed by VM as VM_{Energy}
17 : $VM_{Energy}= VM_{Power} * TEC_{time}$
/* Resource Allocation Phase*/
18: If vm_j status is IDLE && VM_{Energy} is minimum
19: Assign VM_{min}<- vm_j
20: End If
21: Set VM_{min} status as BUSY
22: Allocate VM_{min} to t_{max}
23: End For

Energy Efficient Workflow Scheduling Algorithm

EEWS scheduler flow diagram is depicted in Fig. 3. Energy Efficient Workflow Scheduling in cloud environment consists of two steps such as Task Selection and Resource Selection. The workflow tasks are mapped onto appropriate virtual machines in such a way that the overall resource utilization is maximized and the energy consumption is minimized. The task with maximum length is chosen for the execution. The resource that consumes less energy is selected to map to the task. The energy consumption of a VM is calculated using the power and execution of task.

The power consumption is calculated based on the factors such as power consumed by the VM at idle state, power consumed by the VM when it utilizes full CPU speed and the current VM utilization. The execution time of VM is calculated based on the computation time and the communication time (Fig. 4). The estimated energy consumption of a VM is multiple of the estimated power consumption and the execution time of the task on the selected resource.

========== OUTPUT ==========

Cloudlet ID	STATUS	Data center ID	VM ID	Time	Start Time	Finish Time	Depth
25	SUCCESS	2	0	0.91	0.1	1.01	0
4	SUCCESS	2	3	4.35	1.01	5.36	1
0	SUCCESS	2	1	5.78	1.01	6.79	1
2	SUCCESS	2	0	11.47	1.01	12.48	1
1	SUCCESS	2	2	17.57	1.01	18.58	1
10	SUCCESS	2	1	6.39	12.48	18.87	2
3	SUCCESS	2	4	24.12	1.01	25.13	1
12	SUCCESS	2	1	7.07	18.87	25.95	2
9	SUCCESS	2	4	2.89	25.13	28.03	2
5	SUCCESS	2	2	13.42	18.58	32	2
7	SUCCESS	2	2	2.38	32	34.38	2
8	SUCCESS	2	3	26.9	12.48	39.38	2
13	SUCCESS	2	4	18.5	28.03	46.53	2
11	SUCCESS	2	1	42.9	25.95	68.85	2
6	SUCCESS	2	0	87.99	18.58	106.56	2
14	SUCCESS	2	0	5.95	106.56	112.51	3
15	SUCCESS	2	2	1.8	112.51	114.31	4
19	SUCCESS	2	2	13.85	114.31	128.17	5
18	SUCCESS	2	4	19.17	114.31	133.48	5
20	SUCCESS	2	3	26.31	114.31	140.62	5
17	SUCCESS	2	1	43.43	114.31	157.74	5
16	SUCCESS	2	0	85.86	114.31	200.18	5
21	SUCCESS	2	2	2.73	200.18	202.91	6
22	SUCCESS	2	2	3.84	202.91	206.75	7
23	SUCCESS	2	2	4.89	206.75	211.64	8
24	SUCCESS	2	0	3.78	211.64	215.41	9

Total Energy Consumption :: 37.92917513273758

Fig. 4 EEWS Scheduler Output

5.0 Experimental Results

Performance Comparison – Cost

The Execution Cost comparison for four existing algorithms against CBWS algorithm for Montage application with 25 tasks is shown in Fig. 4 and the behavior for different number of VMs for CyberShake application is shown in Fig. 5.

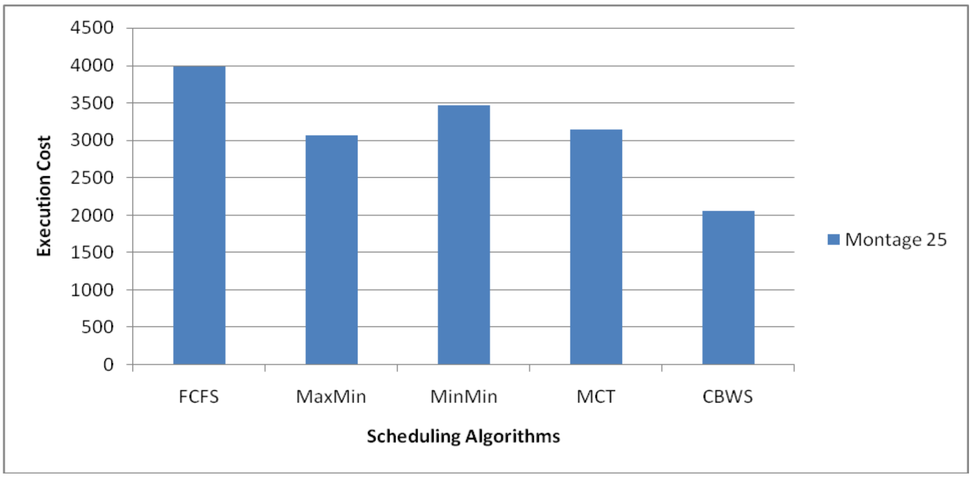

Fig. 5 Comparison on Execution Cost

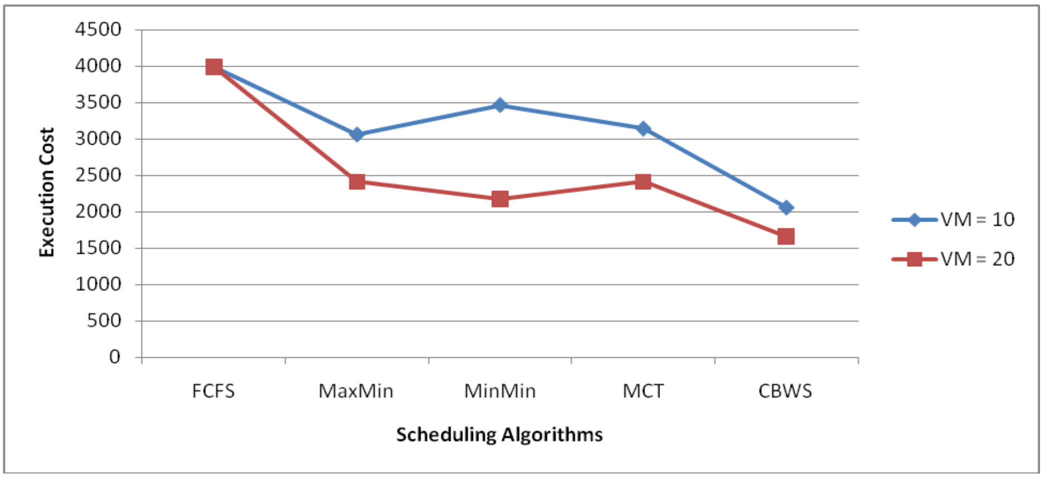

Fig. 6 Comparison on Execution Cost by varying VM count

Performance Comparison – Energy

The Energy comparison for four existing algorithms against EEWS algorithm for Montage application with 25 tasks is shown in Fig. 7 and the variations for different VM count for CyberShake application is depicted in Fig. 8.

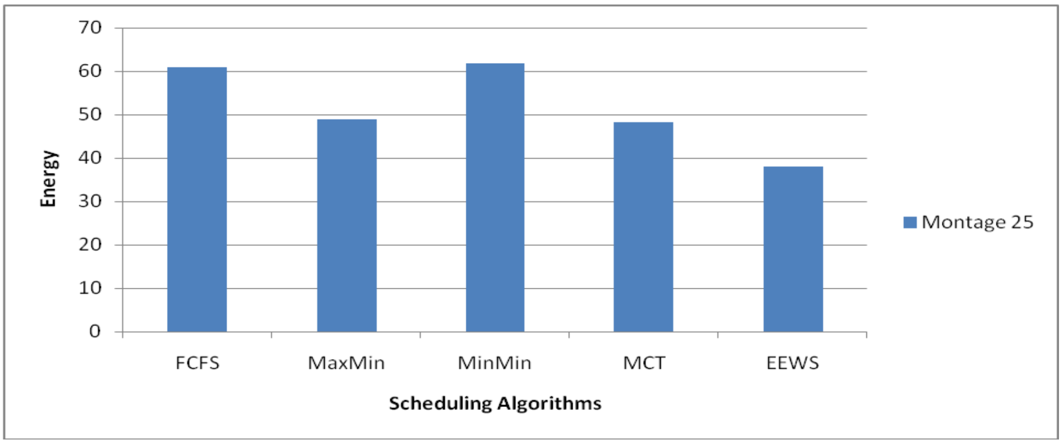

Fig. 7 Comparison on Energy

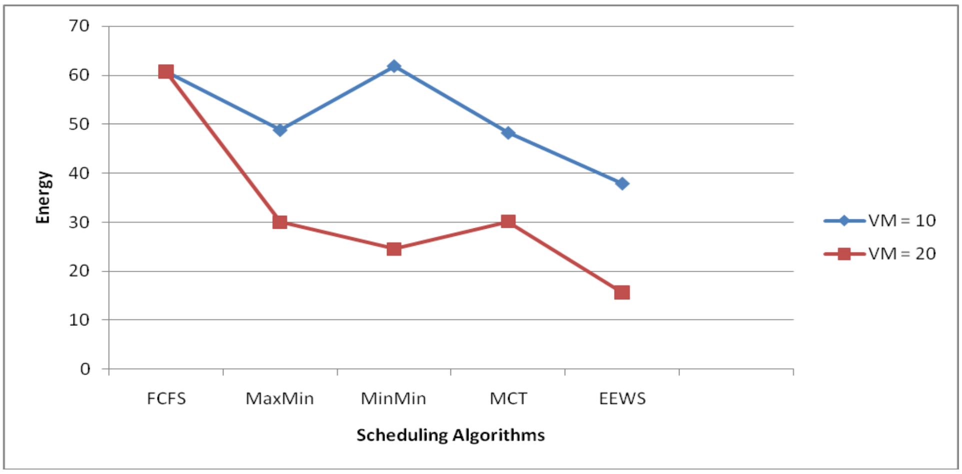

Fig. 8 Comparison on Energy by varying VM count

6.0 CONCLUSION

Workflow Scheduling in Cloud Environment is an interesting area of research gaining much importance in recent research. Effective Scheduling must satisfy the user Quality of Service (QoS) parameters without violating the Service Level Agreement. The proposed work is designed to schedule and allocate tasks to appropriate resources in such a way that cost and energy are reduced and the workflows execute with minimum execution time. Three different workflow scheduling algorithms are presented for assigning the tasks with maximum runtime to virtual machines that are estimated to execute the selected tasks with minimum cost and energy. The experimental results reveal that the proposed algorithms perform better than the existing approaches in terms of specified parameters.

REFERENCES

1. Artem M. Chirkin, A. S. Z. Belloum, Sergey V. Kovalchuk, and Marc X. Makkes. 2014, “Execution time estimation for workflow scheduling”, In Proceedings of the 9th Workshop on Workflows in Support of Large-Scale Science (WORKS’14). IEEE PressPiscataway, NJ, USA, 1-10. DOI=http://dx.doi.org/10.1109/WORKS.2014.11
2. O. M. Elzeki, M. Z. Reshad, M.A. Elsoud, “Improved Max Min Algorithm in Cloud Computing”, In International Journal of Computer Applications, Vol. 50, July 2012.
3. Gupta, M. S. Kumar and P. K. Jana, “Compute-intensive workflow scheduling in multi-cloud environment”, 2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Jaipur, 2016, pp.315-321. DOI: 10.1109/ICACCI.2016.7732066
4. MaciejMalawski, Gideon Juve, EwaDeelman, JarekNabrzyski, “Algorithms for cost- and deadline-constrained provisioning for scientific workflow ensembles in IaaS clouds”, Journal Future Generation Computer Systems, Volume 48 Issue C, July 2015 Pages 1 - 18.
5. Verma, A. & Kaushal, “Cost-Time Efficient Scheduling Plan for Executing Workflows in the Cloud”, S. J Grid Computing (2015) 13: 495. DOI:10.1007/s10723-015-9344-9.
6. Navneet Kaur, Sarbjeet Singh, “A Budget-Constrained Time and Reliability Optimization BAT Algorithm for Scheduling Workflow Applications in Clouds”, Procedia Computer Science 98 (2016), 199 – 204 , DOI: 10.1016/j.procs.2016.09.032
7. Xiaolong Xu, Wanchun Dou, Xuyun Zhang, Jinjun Chen, “EnReal: An Energy-Aware Resource Allocation Method for Scientific Workflow Executions in Cloud Environment”, IEEE Transactions on Cloud Computing, vol. 4, no. , pp. 166-179, April-June 2016, DOI:10.1109/TCC.2015.2453966
8. Guangyu Du, Hong He, and QinggangMeng, “Energy-Efficient Scheduling for Tasks with Deadline in Virtualized Environments”, Mathematical Problems in Engineering, vol. 2014, Article ID 496843, 7 pages, 2014. DOI:10.1155/2014/496843

9. Zhongjin Li, Jidong Ge, Haiyang Hu, Wei Song, Hao Hu, and Bin Luo., "Cost and Energy Aware Scheduling Algorithm for Scientific Workflows with Deadline Constraint in Clouds," DOI: 10.1109/TSC.2015.2466545, IEEE Transactions on Services Computing.

10. Zhuo Tang , Ling Qi , Zhenzhen Cheng , Kenli Li , Samee U. Khan , Keqin Li, "An Energy-Efficient Task Scheduling Algorithm in DVFS-enabled Cloud Environment", DOI: 10.1007/s10723-015-9334-y ,© Springer Science+Business Media Dordrecht 2015.

11. Amandeep Verma , Sakshi Kaushal , "Cost Minimized PSO based Workflow Scheduling Plan for Cloud Computing", In I.J. Information Technology and Computer Science, 2015, 08, 37-43, DOI: 10.5815/ijitcs.2015.08.06 .

12. Deepak Poola, Saurabh Kumar Garg, RajkumarBuyya, Yun Yang and KotagiriRamamohanarao, "Robust Scheduling of Scientific Workflows with Deadline and Budget Constraints in Clouds", 1550-445X/14 $31.00 © 2014 IEEE DOI: 10.1109/AINA.2014.105.

13. Farid M, Latip R, Hussin M, Abdul Hamid NAW, "A Survey on QoS Requirements Based on Particle Swarm Optimization Scheduling Techniques for Workflow Scheduling in Cloud Computing", Symmetry. 2020; 12(4):551. https://doi.org/10.3390/sym12040551