Optimization of Cost and Meeting Deadline in Scientific Workflow

Ruchita P. Pingale[#1], Smita S. Patel[#2]

[#1]*Dept. of Information Technology, Smt. Kashibai Navale College of Engineering, PUNE, India.*

[#2] *Dept. of Information Technology, Smt. Kashibai Navale College of Engineering, PUNE, India.*

Abstract *Cloud computing is booming technology in the area of information technology. Nowadays, the clouds are known as the global storage and used by many companies, schools, websites etc. The elasticity property of the cloud makes it a suitable platform for the execution of the scientific workflow with the deadline constraint. Resource required for the application is dynamically allocated. The existing algorithms in the area of the scientific workflow either try to minimize the cost or focus on minimizing execution time while trying to meet the application deadline. Also, the existing algorithm considers only one data Centre of the cloud. To increase the performance of scheduling process within the deadline we proposed the enhancement to the EIPR algorithm which uses the idle time slots for providing resource and the budget surplus to replicate the task. Replication uses another data Centre for scheduling process. However, the soft deadline is considered for the process. The working of EIPR algorithm checked on the experiments like montage (25, 50- this is the no of task included in the graph). This shows the implemented algorithm (EIPR) is able to minimize the cost and the execution time of scheduling process in scientific workflow.*

Keywords–*Scientific Workflow, Soft Deadline, EIPR, Data Centre.*

I. INTRODUCTION

Cloud computing is delivery of hosted services over the Internet. It is the latest upcoming trend in distributed computing field that gives both hardware infrastructures and software applications as services. Public clouds are mostly used because these infrastructures are available in pay-per-use system. The usage of these services is based on a Service Level Agreement (SLA) which defines their required Quality of Service (QoS) parameters, on a pay-you-use basis [1]. These services can be classified into 3 categories which are - Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). The IaaS Clouds, e.g. Amazon, provide virtualized hardware and storage which facilitates users to deploy their applications and services on it [2]. PaaS Clouds, e.g. Microsoft Azure, provide an application development environment where the users can implement and run applications directly on the Cloud.

Many algorithms are developed for the scheduling process in workflow. But these algorithm are either consider the performance of the workflow in context with the cost minimization or focuses on the minimization of the execution time. The algorithm which is proposed here is focuses on both issues in cloud. For this it considers the soft deadline. Soft deadline is a deadline that, when unmet still investment is not lost as the deadline missed by small margined. The implemented algorithm is ***E****nhanced* ***I****C-****P****CP with* ***R****eplication (EIPR)* algorithm is uses the IaaS cloud partial critical path (IC-PCP) algorithm for minimizing the cost and to get the deadline of the task or process. This algorithm uses the task replication to do so. Task is defined as the one unit of the process. E.g. when the file is uploading is the process then the selection of file, file uploading on net, assigning to particular person are the task. The scientific workflows are used for the experimental study of the data.

Cloud workflow can be referring to the workflow applications which are executed in cloud computing environment. Workflow scheduling is defined as a kind of global task scheduling as it focuses on mapping and managing the execution of interdependent tasks on shared resources like cloud services [6]. Workflow constitutes a common model which describes the wide range of scientific application in the distributed system. Scientific workflows are described as direct acyclic graphs (DAGs) in which nodes represent tasks and vertices represent dependencies among tasks. Workflow Scheduling is a common NP-complete problem which is the process of mapping the workflow tasks to its appropriate resources in the public cloud. Workflows mainly concentrate on the resource automation of procedures which passes the files and data among the participants based on a set of rules. Using scientific workflows numerous complex applications can be broken down into smaller components and that components can be executed reliably and efficiently than original one. The Pegasus Workflow management system is one of the most used workflow of scientific workflow application. This scientific workflow compiles the complex workflows into executable workflow. Following are the workflow scheduling algorithms: Ant Colony optimization Algorithms, Genetic Algorithms, Particle Swarm Optimization (PSO) algorithms, Partial Critical Path (PCP) algorithms [10].

II. RELATED WORK

Cloud workflow is widely applied in scientific research, and the major problem in cloud computing is workflow scheduling. Cloud workflow has the application, platform, and unified resource and fabric concerns as main components of its architecture. In the cloud workflow architecture, the objectives of scheduling the cloud workflow are usually to optimize cost and time. In this survey the all scientific workflow and the different techniques and algorithms used by to get the objectives is described. As cloud computing has the elasticity property and it provides powerful computing power and storage capacity, scientific workflows and business workflows are executing in cloud environments.

Two resources with the same characteristics may have different performance in a given time, what results in variation in the execution time of tasks that may lead to delays in the workflow execution.to avoid this type of delayed and to reduce the impact of performance variation of public Cloud resources in the deadlines of workflows, Rodrigo N. Cahiers[1] proposed a new algorithm, called EIPR, which takes into consideration the behaviour of Cloud resources during the scheduling process and also applies replication of tasks to increase the chance of meeting application deadlines. This algo uses the task replication to increase the performance.

S. Abrishami [2] proposed another algorithm for the other model of cloud like IaaS. Which are uses widely as they were apply on the public cloud. As it have a feature of pay per use. So for this purpose the IC-PCP algorithm is used. This is a one phase algorithm where the deadline of task is achieved. Also this algorithm reduces the execution time in the user budgets. But this algorithm does not work on the fluctuation of the task when task get scheduled. This may lead to increase in the total execution time. Also the variation in performance of cloud workflow cannot handle by this algorithm. In this algorithm first it schedules the critical path and then checks dependencies according to subtasks. For that it uses the entry and the exit tasks. But the problem with this system is this algorithm is unable to find the execution time required for the scheduling. If the resources or the VM are fail for some instance delay may be happen this kind of situation are not handled by the system.

Scheduling of workflows (also referred as Direct Acyclic Graphs DAGs) in parallel and distributed systems has been subject of extensive research. Xin YE [3] has done the survey on the scientific workflows. In which he stated the all cloud types with the services. Paper also proposed the different types of the scientific workflow application. Also paper shows up the all the challenges faced by the workflow scheduling in real world scenario. this paper explains the PCP algorithm. The PCP scheduling algorithm is proposed for the utility Grid model. This has two main phases: Deadline Distribution and Planning. In the Deadline Distribution phase, the overall deadline of the workflow is distributed over individual tasks. this algorithm supports only the software as service platform on the cloud which is also a problem with this system.

To reduce the impact of performance variation of public Cloud resources in the deadlines of workflows, *Ms.K.Sathya,* [4] proposed an existing algorithm, which takes into consideration the behaviour of Cloud resources during the scheduling process and also applies replication of tasks to increase the chance of meeting application deadlines. If the virtual machines fails during the execution then more vm are required to complete the action which may causes the increase in the cost or energy consumption is also high in such cases

To overcome this situation S. Abrishami [6], proposed the Partial Critical Path algorithm. In which the critical path is nothing but the longest path in the workflow scheduling. PCP algorithm is generally works for the QoS. Also the SC-PCP algorithm is proposed for workflow scheduling in SaaS Clouds, which minimizes the total execution cost while meeting a user-defined deadline. But these algorithms are only applicable for SaaS layer or model. These algorithms cannot works on other models like PaaS and IaaS.

Nallakumar R. [7] is surveyed on the workflow scheduling. The direct acyclic graph use to represent the workflow in cloud. In this paper all the algorithms used for the scheduling of the scientific workflow are stated. These algorithms are based on the deadline constraints but they minimize the total cost. These algorithms are either minimizes the cost or focuses on the deadline of the task hence these algorithms cannot works on the fluctuation of the task in the workflow.

To know the resources need for the task Eun-KyuByun [8] evaluates the algorithm PBTS for estimating the minimum resource capacity to execute a workflow within a given deadline. Synthetic and real workflows have demonstrated that the PBTS algorithm performs better than the alternative approaches in terms of cost, and its performance is close to the theoretical low bound. This system is crashed if task from the PCP delays.

To avoid the variation in the performance of workflow the soft deadlines are considered. KassianPlankensteiner [9] introduces the heuristics which uses the task replication and task resubmission to increase the performance of task shading. New heuristic uses the soft deadline for workflow execution. This system uses the resubmission techniques. In this technique the whenever there is failure occurred then the whole task is resubmitted to the VM and start the process from the beginning. The only problem of this algorithm is the resubmission will increase the execution time and the delay may be occurred. And also this algorithm never predicts the failure of resources. If failure happens it simply does the resubmission.

In another paper[10] propose the LBMMC algorithm This methodology is used for the migration purpose. Migration does the load migration from one virtual machine to another virtual machine. This helps to minimize the time use for scheduling task. The problem with this system is the migration of the task increase the cost and also it may lead to creating the decencies as some tasks are migrated to another VM. Also this system does not support the physical environments. It supports only the physical environments.

III. SYSTEM ARCHITECTURE

In this paper, the proposed system is implemented for minimizing or meeting the deadline of the task in workflow scheduling process performed on the cloud. To meet the deadline and to minimize the cost we are implementing task replication in the scheduling process. In the proposed method the problem statement is decomposed in different modules like analysis of virtual machines, assigning the deadline, execution of replica, task submission.

This system implements the EIPR algorithm and also uses the FCFS (first come first serve) algorithm for the scheduling. Fig 1 shows the system architecture diagram, which evolved the different module in system. First of all it checks for the login user's permission and allow them to upload. After the successful login user can do their task and submit it. The database keeps the record of users and their permission for access. When the any task done by the user e.g. downloading of file from the cloud then it check for dependencies i.e. if file is not available then discard the process and shows error message to the user. Also the deadline for the task is also check at the same time. The replicas can be created and schedule on the different virtual machine if the time taken by the tasks is greater than default deadline of particular task.

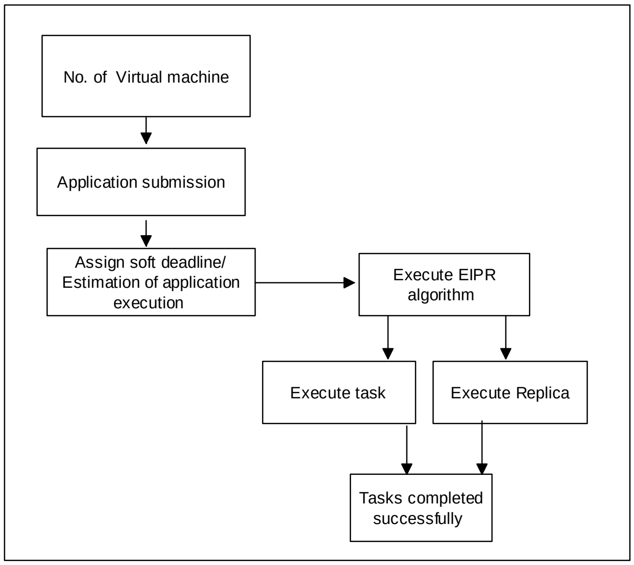


Fig 1. System Architecture

The working of the different module of the system is explained below.

- No. of Virtual Machine: In this module the total number of the virtual machines are calculted where the tasks or the process can be scheduled in the cloud.it simply checks the available virchual machine (VM) in cloud VM list.
- Application Submission: The process done by the user on cloud i.e. uploading, downloadling, or storage is submitted by user. and the repective action are taken (as user is uploading the file then encryption will done). For submission of the application user must be an authenticated to do such task.
- Estimation of Application Execution: the total time requored for the completion of task is estimated. And check with the deadline assign for the task. this results are stored in the database or furthure use.
- Execution of Task: in this module the task is actually hosted on the cloud. the time taken by the task is calculated and compare with the estimated deadline. if comaprision result is positive then it just finishes the exection. else replication part will be executed.
- Execute Replicas: replicas of the current task is created and schedule on the different virtual machine for the execution. this helps to get the deadline and also if for any reason if one VM is failed to complete its task then system can get results from the replicated task which is scheduled on different virtual machines. Time taken by the task and replica also compare for furture scheduling process, VM with better perfomace is selected.

IV. RESULT AND DISCUSSIONS

There are many scientific workflows like montage, ligo, cybershake etc. we are testing our algorithm with montage (having any number of tasks-25,50,100etc) standard workflow. The As following table and the graph shows that the performace of the EIPR algorithm having replication of task.

Total Number of Violations in the application Deadlines after Execution of 50 Instances of Each Application. The Budget on EIPR Represents the Extra Budget Available for Replication, in Relation to the Amount Spent before the Replication.

The generic and the EIPR algorithm are work on same issue i.e deadline of workflow and the cost but EIPR algorithm has better performance than the generic algorithm. Following graph shows the total execution time for the both generic and the EIPR algorithm.

	Genetic Algorithm(in ms)	Proposed EIPR(in ms)
Task1	700	400
Task2	650	380
Task3	980	700
Task4	1020	800
Task5	1000	750

Table 1:Shows execution time Genetic algorithm and Proposed Enhanced IC-PCP with Replication (EIPR) algorithm.

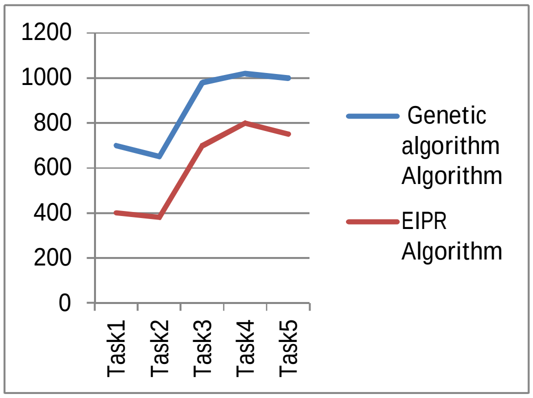
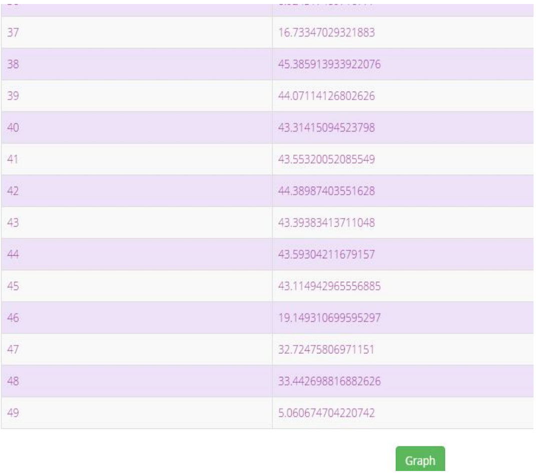


Fig.2 Shows execution time of each task. X-axis Tasks , Y-Axis Time in ms of Genetic And proposed EIPR algorithm.

As shown in the graph x-axis represts the total execution time for the task in the workflow and y-axis shiws the cost requied for each task in the workflow.



37	16.73347029321883
38	45.385913933922076
39	44.07114126802626
40	43.31415094523798
41	43.55320052085549
42	44.38987403551628
43	43.39383413711048
44	43.59304211679157
45	43.114942965556885
46	19.149310699595297
47	32.72475806971151
48	33.442698816882626
49	5.060674704220742

Fig3(a). shows the cost for each task id

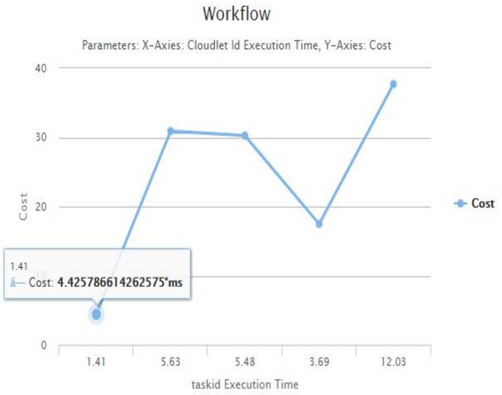


Fig3(b). X-axis: execution time Y-axis: cost from fig3(a).

V. CONCLUSIONS

The provisioning and scheduling property of cloud infrastructures makes it a suitable platform which can be execute the workflow applications considering the soft deadlines. Many times fluctuation is occurred in the public clouds while actual performance delivered by resources. To reduce the impact of performance variation of cloud resources in the deadlines of scientific workflows, here a new algorithm, called EIPR is used. This useful for the behaviour of cloud resources during the scheduling process and also applies the task replication to increase the chance of meeting application deadlines.

ACKNOWLEDGMENT

On the submission of this paper, I would like to extend my gratitude & sincere thanks to my project guide Prof. S. S. Patel, of Information Technology for her constant motivation and support during the course of my work. I truly appreciate and value her guidance and encouragement. I would like to thank all faculty members of SKN College of Engineering and all who's direct and indirect support helped me in my research work of paper.

References

[1] Rodrigo N. Calheiros, RajkumarBuyya, "*Meeting Deadlines of Scientific Workflows inPublic Clouds with Tasks Replication*"SYSTEMS, VOL. 25,pp-1787-1796 , JULY 2014.J. Breckling, Ed., *The Analysis of Directional Time Series: Applications to Wind Speed and Direction*, ser. Lecture Notes in Statistics. Berlin, Germany: Springer, 1989, vol. 61.

[2] S.Abrishami, M. Naghibzadeh, and D. Epema, ''*Deadline-Constrained Workflow Scheduling Algorithms for IaaSClouds*,''FutureGener. Comput. Syst., vol. 29, no. 1, pp. 158-169, Jan. 2013.

[3] Ms.K.Sathya, Dr.S.Rajalakshmi,"Deadline Based Task Scheduling in Cloud with Effective Provisioning Cost using LBMMC Algorithm",Volume 1 Issue 7, November 2014.

[4] Xin YE,JiweiLIANG,SihaoLIU,Jia LI,"*A Survey on Scheduling Workflows in Cloud Environment*",International Conference on Network and Information Systems for Computers, pp.344-348, November 2015. (2002) The IEEE website. [Online]. Available: http://www.ieee.org/

[5] Ms. B. Poornima,Prof. S. R. Mugunthan,"*Meeting Deadlines Constraint of Scientific Workflows in Multiple Cloud by Using Task Replication*",(ICSNS -2015), Feb. 25 – 27, 2015.*FLEXChip Signal Processor (MC68175/D)*, Motorola, 1996.

[6] S. Abrishami, M. Naghibzadeh,"*Deadline-constrained workflow scheduling in software as a service Cloud*",FutureGener. Comput. Syst., vol.19, 680–689, Nov.2012.

[7] G. Juve, A. Chervenak, E. Deelman, S. Bharathi, G. Mehta, and K. Vahi, "*Characterizing and Profiling Scientific Workflows*,'' Future Gener.Comput.Syst., vol. 29, no. 3, pp. 682-692, Mar. 2013.

[8] Nallakumar. R1, SruthiPriya. K. S2 "A Survey on Deadline Constrained Workflow Scheduling Algorithms in CloudEnvironment" (IJCST) – Volume 2 Issue 5, Sep-Oct 2014,pp 44-50.

[9] Eun-KyuByun, Yang-Suk Kee, Jin-Soo Kim, SeungryoulMaeng,"*Cost optimized provisioning of elastic resources for application workflows*",Future Generation Computer Systems 27 pp. 1011–1026,may 2011.

[10] KassianPlankensteiner,RaduProdan,"*Meeting Soft Deadlines in Scientific workflows Using Resubmission Impact*",PARALLEL AND DISTRIBUTE SYSTEMS, VOL. 23, NO. 5, MAY 2012