Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3
© 2005 - 2014 JATIT & LLS. All rights reserved·

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

EFFICIENT SCHEDULING OF WORKFLOW IN CLOUD ENVIORNMENT USING BILLING MODEL AWARE TASK CLUSTERING

D.A.PRATHIBHA[1], B.LATHA[2] AND G. SUMATHI[3]

[1]Department of IT,Sri SaiRam Engineering College, Anna University,Chennai,India.
[2]Department of CSE,Sri SaiRam Engineering College, Chennai,India.
[3]Department of IT ,Sri Venkateswara College of Engineering,Anna University,India.
E-mail:[1]Somaprathi25@gmail.com

ABSTRACT

Cloud computing is a cost effective alternative for the scientific community to deploy large scale workflow applications.For executing large scale scientific workflow applications in a distributed hetereogenous enviornment ,scheduling of workflow tasks with the dynamic resources is a challenging issue.Moreover in a utility based computing like cloud which supports pay per use model of the resources ,scheduling algorithm must efficiently utilize the available time of the resource.Most of the existing scheduling heuristics does not consider the dynamic nature of the cloud and hence produce the static schedule. Public cloud enviornment like Amazon EC2 offers catalog of resources and the price is generally metered per hour.Here any fractional usage is rounded off to the next hour.To meet the budget and deadline of the customers proposed work focuses to incorporate a billing model aware task clustering mechanism in the workflow scheduling process . This work also presents a resource selection algorithm which can be used for choosing proper resource at each stage in the workflow. Preliminary results obtained by running two scientific applications Montage and Cybershake with different resources and task clustering mechanisms are discussed.

Keywords: *Cloud Computing,Workflow,Resource Selection,Deadline,Budget,Task Clustering*

1. INTRODUCTION

Cloud computing is transforming enterprise IT Infrastructure design.It is the alternative to the organizations looking for the goal of building flexible,low cost and scalable services which can be accessed over the internet.Cloud computing is the large scale distributed computing paradigm in which pool of abstracted ,virtualized,dynamically scalable computing resources and services can be accessed on demand over the internet.The services of the cloud are primarily provided at three levels:IaaS(Infrastructure as a service),Platform as a Services(PaaS) and Software as a Services(SaaS).The goal of cloud computing is also same as other hetereogenous distributed computing platforms grid and cluster.The objective of these distributed computing paradigm is to provide unlimited access to the powerful computing resources.Cloud computing extends this objective by providing metered services .

Recently there has been great interest in applying cloud computing technology to solve large scientific and business applications[1] which consists of thousands of tasks with huge number of computations and data transfer. The tasks in these applications are executed in a certain predefined order.One of the challenge for the scientific community is to provide powerful and efficient programming model to represent the scientific applications .These applications are modeled as workflows and are used to solve various problems in the areas like astrononmy, bioinformatics,weather monitoring ,earthquake science. The primary benefit of moving workflow applications to clouds is application scalability. Unlike Grids, scalability of Cloud resources allows real-time provisioning of resources to meet application requirements at run-time or prior to execution. The elastic nature of clouds facilitates changing of resource quantities and characteristics to vary at runtime, thus dynamically scaling up when there is a greater need for additional resources and scaling down when the demand is low.

Journal of Theoretical and Applied Information Technology
31st July 2014. Vol. 65 No.3
© 2005 - 2014 JATIT & LLS. All rights reserved·
ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

One of the main reason for running scientific workflow applications in distributed systems like cloud is to execute the workflow with short execution time in less cost.This emphasis the need for optimization of scientific workflow so that the customer and service provider are mutully benefited.Workflow scheduling and dynamic resource allocation in cloud platform plays very important roles in this optimization process.Task to resource mapping is widely addressed in grid computing with many scheduling heuristics have been developed. Unlike in grid ,in cloud environment this process is very complex and has to deal with dynamic resources and also with the unique billing model .

This paper is organized as follows section 2 describes Workflow concepts . In section 3 related work of workflow scheduling in cloud is presented. Section 4 problem statement is established followed by section 5 in which prelimnary results are discussed and section 6 gives conclusion and future work in this research area.

2. WORKFLOWAPPLICATIONS

Scientists in various research fields, work with complex applications and conduct experiments which require huge computational power, large memory, high speed inter connected networks which are typically offered by super computers or HPC clusters. Scientific applications will have large number of tasks which are interdependent. Many tasks also require parallel execution in order to obtain high performance.

2.1 Workflow Modelling.

The structure of the workflow indicates the order of execution of the tasks. Based on the representation workflows can be classified as directed acyclic graph (DAG) or a non-DAG. In DAG-based workflow, a graph G {V, E} in which vertices V = {T1... Tn} denotes the individual task of the workflow and edges E of the graph denotes a task dependency relationship between the nodes. The DAG also represents the precedence constraints among the tasks i.e. for each (t_i, t_j) ϵ E, t_j must be executed after the end of the execution of t_i.

Fig.1 shows the DAG representation of workflow, where A, B, F represents a sequence and B, C, D represents parallelism. In addition to all structures contained in a DAG-based, a non-DAG workflow also includes iteration structure, in which sections of workflow tasks in an iteration block are allowed to be repeated. A task serving a specific function may process large amount of data. Many of the workflow applications used by the scientific community in fields like Astronomy, Weather Monitoring, Bioinformatics applications are running on supercomputers. As the amount of data increases exponentially distributed environments like cluster, grid and cloud computing are also suitable for deploying workflow applications as they offer heterogeneous environment.

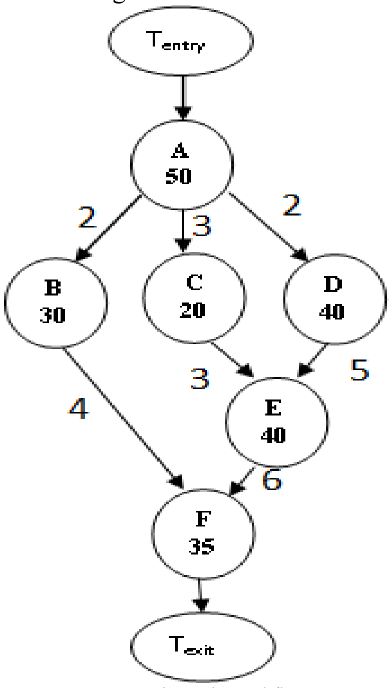


Figure. 1. DAG-based Workflow representation

Two dummy tasks T_{entry} and T_{exit} with zero execution time are used to indicate the beginning and ending of the workflow For any task T_i ϵV W_{ij} is defined as the execution time of T_i on resource R_j .Average execution time of task T_i on 'm ' heteregenous resources $\overline{w_i}$ can be computed by the following equation

$$\overline{w}_i = \sum_{i=1}^{m} \frac{w_i}{|m|} \tag{1}$$

Every edge (T_i, T_j) in E, is associated with value tr_{ij}, representing the time needed to transfer data from T_i to T_j.The transfer time can be calculated according to the bandwidth $b_{x,y}$ between the resources executing these tasks p_x and p_y respectively.Data transfer time between two tasks is zero if they are deployed on same resource.

$$\mathrm{tr}_{ij} = \frac{d_{ij}}{b_{x,y}}$$

(2)

In general the execution costs (ec) and transmission cost(tc) are inversely proportional to the execution times and transmission times respectively.Overall execution cost for deploying the workflow in a heterogenous enviornment is given by

Total_cost=exec_costs(ec)+trans_costs (tc) (3)

2.2 Workflow and Task Clustering

In task clustering, small tasks are grouped together as one executable unit such that overhead of data movement is eliminated and also improving the deadline. Pandey et al. [2] ,proposed clustering of tasks based on their execution time, data transfer and level. If tasks were having high deviation and value of average execution time, they were executed without clustering. Tasks with lower deviation and value of execution time were clustered together. The results indicate improvement in makespan for data intensive workflow applications. However the side affect of task clustering is it may result in more failure rate since the job contains more than one task.These failure rates can have significant impact on the performance of the workflow.A framework for task failure model and job failure model that addresses these performance issues in task clustering is proposed[3] . We continue to enhance the existing work in task clustering with the billing model of the public cloud.

3. RELATED WORK

The workflow scheduling problem in cloud as in other heterogenous computing systems is also an NP-hard optimization problem, i.e., the amount of computations needed to find optimum solution increases with the problem size. The most widely used heuristics for scheduling the workflow application is Heterogeneous Earliest Finish Time (HEFT) algorithm developed by Topcuoglu et al. [4]. It is a static scheduling algorithm that attempts to minimize makespan. In [5], the authors propose an extension to HEFT by addressing the elasticity nature of the cloud and propose Scalable-Heterogeneous-Earliest-Finish-Time in which the resources are ‘scaled in’ if there is a resource whose available time is equal to the minimum finish time of a given task otherwise the resources are scaled out.

In [6] authors propose Balance time scheduling algorithm, which computes minimum number of resources required, considering the idle time of the resource in each iteration. In[7] authors extended their previous work and proposed the Partitioned Balanced Time Scheduling (PBTS) algorithm for cost-optimized and deadline-constrained execution of workflow applications on clouds. Limitation of this approach is that it considers only one type of cloud resource, which would has been decided in advance.

The upgradation fit algorithm which is based on the make span of the application either does vertical or horizontal optimization[8]. In vertical optimization the tasks are combined and they are tested whether they are compatible with high end virtual machine. In Horizontal optimization minimizing the number of VMs by using the Best Fit algorithm is done .In [9] progress share algorithm is used for resource allocation in order to have a fair utilization and also introduce job affinity for selecting a resource. In [10] repetitive execution of scientific workflow application is considered and Provenance-based adaptive scheduling heuristic for parallel scientific workflows in cloud is proposed. This schedules the task based on three factors: cost, deadline and reliability.

Enhanced IC-PCP with Replication (EIPR)[11] algorithm is proposed which increases the likelihood of completing the execution of a scientific workflow application within a user-defined deadline in a public cloud. It considers the behavior of the cloud resources during scheduling process.

In[12] Partitioning of workflows which are very large and data intensive to reduce the complexity of the workflow is proposed. In the partitioning process the cross dependency among the tasks are checked so as to avoid deadlock loops. The overall workflow execution process consists of three components partitioning, estimator and scheduler. Amazon EC2 has introduced Spot VM instances in which the resources are offered through bidding process .In [13] their work for deploying non real time applications like scientific data analysis use spot VM .Much of the existing work does not consider the dynamic and billing model of the cloud computing for scheduling.

Existing research work on workflow scheduling in cloud, has given limited importance to the dynamic nature of the cloud and also billing time calculation of resources usage of the cloud. This work focuses on enhancing the task clustering mechanism by considering the remaining available time criteria for choosing the resource at each level.A task in a workflow can be computation intensive,data intensive or I/O intensive. Cloud providers come with catalog of resources(small ,medium and large) which vary in performance and cost, hence choosing a proper resource for a given task is a very important issue that need to be addressed. Proposed work focuses on analyzing the importance of task clustering in the execution of workflow.

4. PROBLEM STATEMENT

A cloud environment consists of data centres. It provides resources to the customers on demand for a requested amount of duration. A data center consists of large number of physical machines which are virtualized using a hypervisor, to provide infinite number of virtual resources to the customers. These are known as virtual machines. A public cloud comes with various types of resources which vary in performance and cost. For Example, Table [1] shows the instance types offered by Amazon EC2. Here, the price is generally metered per hour and any fractional usage is rounded off to the next hour, which is considered as BTU (Billing Time Unit). For example, if a VM is used for 75 minutes, then the billing will be done for 2 hours, while the actual resource would have been used only for 1 hour, 15 minutes.Our work focuses on the complete utilization of the resource. This enables the customer to pay only for the resource which was utilized.

The objective of the proposed work is efficient mapping of tasks of workflow to the virtual machine such that it satisfies the given budget and deadline thereby minimizing the SLA violations. In IaaS clouds the cost of running a workflow is mainly the cost of using three main resources storage, compute and network

Table 1:Instance types in Amazon EC2(source: aws.amazon.com/ec2/instance-types)

Instance Type	Memory(GB)	Disk (GB)	No.of Cores	$Hour
m1.small	1.7	160	1	.085
m1.large	7.5	850	2	0.34
m2.2xlarge	34.2	850	4	1
c1.medium	1.7	350	2	0.17
cc1.4xlarge	23	1690	8	1.6
cg1.xxlarge	23	1690	8	2.1

$$\text{Resource\_cost} = \text{Cost(Compute)} + \text{Cost(network)} + \text{Cost(Storage)} \qquad (4)$$

Where

$$\text{Compute\$}(C) = \lceil \text{Cost[VMType]} * \text{hrs} \rceil$$

$$\text{Network\$}(C) = \lceil \text{Cost[per\_hour]} * \text{hrs} \rceil$$

$$\text{Storage\$}(C) = \frac{(\text{Cost[per\_month]} * \text{storage\_size})}{\text{Month-Hrs}}$$

Fig[2] represents the phases in execution of workflow in a heterogeneous cloud system. Many of the workflow management systems provide user interface through which user provides the details of the tasks and the interdependencies .Workflow construction is used to construct the DAG representation which can be described using XML file.In the next phase, before scheduling, workflow is parsed to check for combining the independent tasks to form cluster so that queuing delay can be minimized. Partitioning also can be considered for parallel execution. Next the workflow is given as input to the workflow scheduler, it is responsible for task to resource mapping, resources provisioning, selecting the proper resource . Advanced schedulers may also consider finding usage choice between on-demand, reserved and spot instances for cost optimization. Our work focuses on selecting proper resource and inclusion of billing model of the cloud with task clustering in order to meet cost and deadline specified.

Workflow restructuring techniques like task clustering,replication,partitioning are applied widely in the execution of large scale applications Task clustering mechanism involves grouping multiple tasks into a cluster and execute it as the single. Task clustering improves the response time by reducing the waiting time of the individual tasks. Workflow management systems like Pegasus[14] currently implements level and label based clustering. In level-based clustering, tasks at the same level can be clustered together as shown in Fig[3].

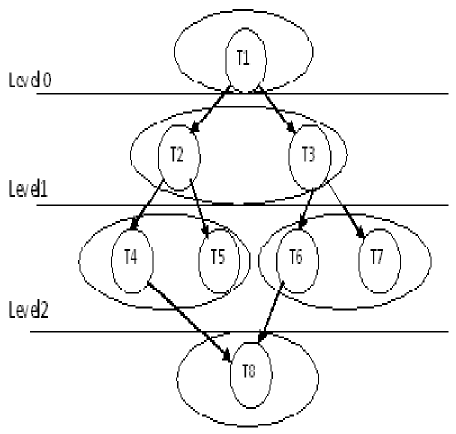


Figure.3 Level based clustering

4.1 Enhanced Task Clustering Mechanism

The existing task clustering does not consider the task execution time and available time of the resource. In the proposed work, the grouping of task is done in each level by calculating the remaining available time of the resource and identifying the suitable combination of tasks that can be mapped to the existing resource.Fig[4] shows example workflow and scheduling process for level based clustering and our proposed model.It can be seen from the diagram that level based scheduling will result in idle time of the paid resource as 110 min and it requires minimum of 4 virtual machines, whereas in the proposed model same workflow can be deployed with 2 VM and it will have only 20 mins as the idle time .

In a public cloud environment, there are diverse type of resources available. In our proposed task clustering, grouping of tasks based on the available time of the resource and task execution time is compared for mapping.If it is not possible with the available resource then a request for new resource is made .In the resource selection algorithm, for each task in each level of DAG workflow resource is chosen such that the workflow is executed within given deadline and budget.

Procedure Deadline and budget distribution (G,D,B)

```
Input: Let T be the set of tasks in the workflow G
    Let R be the set of descriptions of all the available instance types. Deadline D and Budget B.
Output: Obtain Schedule S such that it meets
        Deadline D and Budget B .
Step1: Distribute the Given deadline and budget across each level.
    For ( i=0;i<=Depth(G);i++)
      Deadline_level[i] =distribute_deadline(G,I,D);
      Budget_level[i] =distribute_budget(G,I,B);
    End Loop
    Integer I,makespan=0,cost=0,Level=0;
Step2: Identify the order of execution of dependent tasks.
    Topologicalsort (G)
Step 3: Identifying the proper resource in order to get acceptable deadline and budget.
    For i=0 to i<=levels of the DAG G do
     Resource_selection (G(T[i],R)
Step 4: Modify the resources in the level with higher execution rate.
    If Deadline_level[i]<Current_deadline_level[i]
    then
    Call Res_selection procedure
    End if
Step 5: Modify the resources in the level with low cost available resources
    If Budget_level[i]<Current _budget_level[i]
    then
    Call Res_selection procedure
    End if
End loop
End procedure.
```

Procedure Res_Selection(Graph G,Resource R)

```
Input: T set of independent tasks on each
       level of workflow graph G
       R set of all available resources in cloud.
Output: Mapping(t,r) where t represents the cluster of individual tasks and r is the resource.
Integer: Bag_Tasks[i][m] // i is level and m is number of independent tasks
Integer size[i] //where i is level and size[i] represents number of tasks in that level
Integer Tot_exetime// Cumulative execution time of all the tasks in a given level
For i=1 to Depth(G) do
    Ifthis.task.parent<=1then//
      add(Bag_Task[i][m],this.task)
      increment m
    End if
    Size[i]=m
End loop
For i=1 to Depth(G) do
    Total_exetime =0;
    For j=1 to size[i] do
    Begin
    Tot_exetime = Total_exetime
```

```
                    +this.bag_task[i][j].exetime
    End
End
If Tot_exetime < r.avail_time && r.type =
'small' then
If r.cost=given_cost then
    Execute Bag_Task[i][m] in small instance
 End
Else if r.type=='medium' then
If r.cost=given_cost then
Execute Bag_Task[i][m] in medium instance
End
Else if r.type=='large' then
If r.cost=given_cost then
Execute Bag_Task[i][m] in large instance
End if
End Loop
 End Procedure
```

5. PRELIMINARY RESULTS AND DISCUSSIONS

Simulation is one of the most popular evaluation methods in scientific workflow studies.Performance evaluation for the proposed work has been done with WorkflowSim[15] and CloudSim[16]. Developed by Dr. Chen from University of Southern California, WorkflowSim is able to provide required functionalities and is widely used to simulate workflows in cloud, which is also continuously used to evaluate our proposed methods.Fig[5] shows the layered architecture of the simulator. WorkflowSim consists of a Workflow Mapper to map abstract workflows to concrete workflows that are dependent on execution sites, a Workflow Engine to handle the data dependencies and a Workflow Scheduler to match jobs to resources. WorkflowSim is an extension of CloudSim and it provides additional layer for managing workflow in cloud environment.Experiments were run on Dell machine with Intel(R)Core(TM) i5-3210MCPU@2.50GHz,500GB hard disk storage space and 4GB RAM.

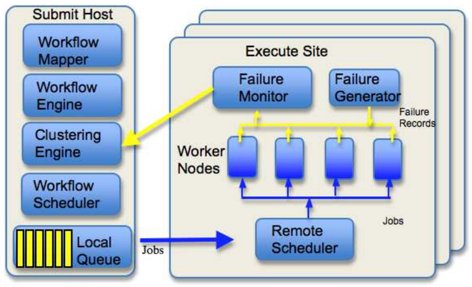


Figure.[5]. An overview of WorkflowSim (source: http://www.workflowsim.org/)

The various instances of virtual machines used in our experiments is given in Table 2.

Table 2 : Resource types used in the experiment

Sl.No	Resource Type	Processing Speed(MIPS)	RAM(GB)	Cores	Bandwidth
1	Small	4000	8	4	2000
2	Medium	8000	16	8	2200
3	Large	16000	64	16	2500

Two workflows are used in the experiment :Montage and Cybershake .Both the workflows are generated using workflow generator. Two experiments are conducted in this work.Experiment 1 is conducted to investigate the impact of resource selection for a workflow in a cloud environment.Montage workflow with different sizes is executed with different resource type and size.Experiment 2 evaluates the performance of workflow(makespan) execution with and without task clustering .Cybershake workflow with different task set is run with various resource types and task clustering techniques.

5.1 Experiment 1:Resource Type selection

This section discusses the experimental evaluation of the significance of resource type on execution of a workflow. The parameters considered are makespan and total cost.

The first application considered in the study experiment is Montage[17] , creates science-grade astronomical image mosaics using data collected from telescopes. It is created by NASA/IPAC Infrared Science Achieve as an open source tool kit ,the custom mosaic is generated by giving input in the format of Flexible Image Transport System. The size of a Montage workflow depends upon the area of the sky (in square degrees) covered by the output mosaic. In our experiments we have considered four cases with number of tasks as 25,50,100 and 125. Montage is an example for I/O-bound workflow applications because the tasks wait most of the time in completing I/O operations.

Fig.6 shows the execution time taken by each Montage workflow when executed on different types of resources like Small, medium and large. From the Figure, it can be observed that irrespective of the workflow size, higher resource capacity yields shorter makespan.

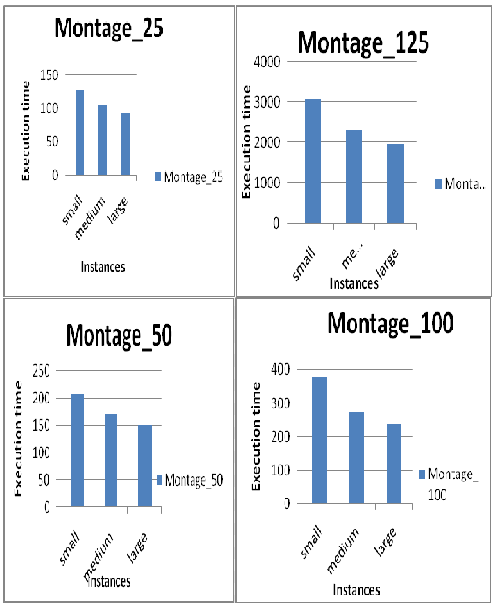


Figure. 6:Montage Workflows Execution On Different Resource Types

5.2 Experiment 2:Performance analysis of workflow execution with task clustering

To efficiently deploy workflow applications in a dynamic environments like cloud,runtime optimization techniques like task clustering can reduce the scheduling overhead and thereby improve the makespan of the workflow.However the existing clustering techniques do not consider the available resource time for scheduling.Proposed task clustering mechanism enhances the horizontal and othere level based clustering technique to accommodate the pay per use feature of the cloud,particulary in public cloud the billing of the resources is done for integer hours(fractional hours are converted to intger hour)to reduce the paid idle time of the resource.

The goal of the experiment is to check the performance of cybershake workflow when it is executed with and without clustering.CyberShake is a seismology workflow application and it is used by the Southern Calfornia Earthquake Center to characterize earthquake hazards in a region. CyberShake utilizes 3D simulations and finite-fault rupture descriptions to compute deterministic (scenario-based) and probabilistic seismic hazards in Southern California [18].Cyber shake workflow consists of two phases .Strain Green Tensor is the first phase which consists of large number of MPI jobs used for wave propogation simulation.Post processing is the second phase with large number of serial jobs after the preprocessing. The size of the workflow depends on the number of sites and frequency.For each site cybershake workflow has two large MPI jobs and 840,000 embarrassingly parallel post-processing jobs

In our experiment we have considered four different sizes 30,50,100 and 1000 of cybershake workflow. Since Cybershake workflow contains large number of parallel tasks the clustering methods horizontal and block are used for comparative study.

Fig[7] shows the distribution of makespan in the 5 levels of cybershake_30 .The results are obtained with number of Virtual machines equal to 10.This analysis will be helpful for deciding how many resources need to be allocated to each level to meet the deadline and budget. Fig[8] shows the cost of executing cybershake_50 with different resource types small,medium and large.For each resource type ,workflow was executed with different sizes of resources.

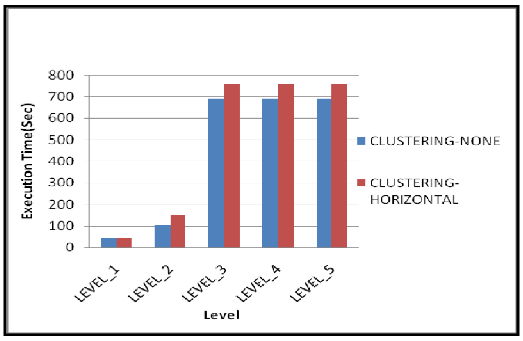


Figure. 7 :Execution Time In Each Level Of Cybershake_30.

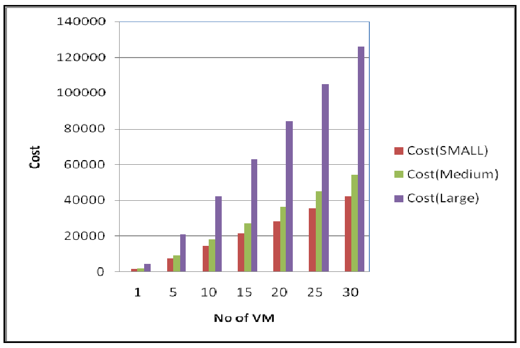


Figure.8: Cost Comparision Of Cybershake_50 WithDifferent Resource Types.

Fig[9] shows the runtime variation in cybershake workflow with tasks 1000.Initially Cybershake_1000 was executed without clustering then it was executed with clustering methods horizontal and block.Clustering size of two was used in the experiment.It may be observed that task clustering works when there are more number of VMs then number of tasks.

In task clustering, the clustering factor (*k*) is defined as number of tasks in a clustered job.Performance of the clustered workflow depends on the total number of jobs in the workflow and it can be calucalated as

Number of jobs=Number of tasks/cluster size.

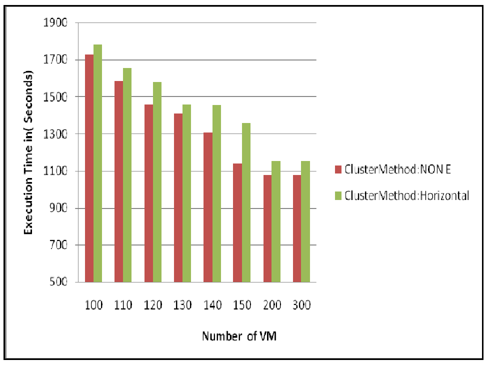


Figure.9:Runtime Comparison Of Cybershake 1000 With Different Clustering Techniques.

From the equation it is clear that if the cluster size is very small ,number of jobs will be very high hence execution time increases.If cluster size is large it results in small number of jobs and in this case the result depend on the type of the job, if the job has large number of small parallel jobs then the overhead will be less .

Fig[10] shows the execution of the cybaershake_1000 workflow with different values of cluster size and number of clusters.It can be observed from the figure that choosing proper k value has impact on the makespan of workflow

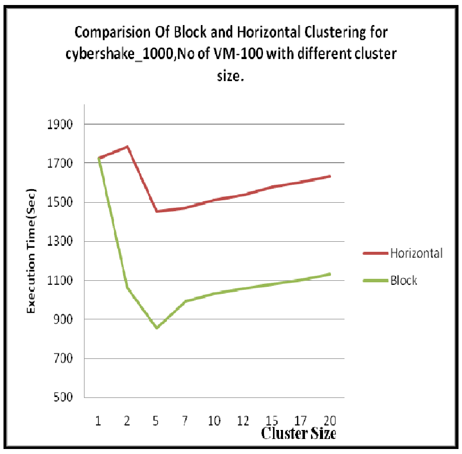


Figure 10: Execution Of Cybershake_1000 With Different Cluster Size

6. CONCLUSION AND FUTURE WORK

In this paper we have proposed efficient workflow scheduling algorithm.It considers the heterogeneous nature of the cloud and billing model of the cloud in the resource selection phase of the workflow scheduling.In this paper enhanced task clustering mechanism is proposed which uses grouping of the tasks on to a resource by considering the execution time of task and available time of the resource.The main rationale is to reduce the paid idle time of the resource in a public cloud,which occurs in converting fractional resource usae to integer hours. Preliminary results of executing two workflows Montage and Cybershake with different resource configurations and clustering mechanism is analyzed for cost and execution time.In future we will work on real time implementation of our proposed system.

REFERENCES

[1] Ewa Deelman, “Grids and Clouds : Making Workflow Applications Work in Heterogeneous Distributed Environments”, International Journal of High Performance Computing Applications, 2009.

[2] S. Pandey, W. Voorsluys, M.Rahman, R Buyya, J. E. Dobson, and K. Chiu, “A Grid Workflow Environment For Brain Imaging

Analysis On Distributed Systems", Concurrency and Computation: Practice & Experience,Vol. 21,2009,pp. 2118–2139.

[3] Weiwei Chen and Ewa Deelman,"Fault Tolerant Clustering in Scientific Workflows", In IEEE International Workshop on Scientific Workflows (SWF), in conjunction with 8th IEEE World Congress on Services, Honolulu, Hawaii, 2012 .

[4]H.Topcuoglu, S.Haririand M. Wu, "Performance Effective and Low-Complexity Task Scheduling For Heterogeneous Computing" , IEEE Transactions on Parallel and Distributed Systems, Vol. 3,2002,pp. 260-275.

[5] Cui Lin and Shiyong Lu,"Scheduling Scientific Workflows Elastically for Cloud Computing", In IEEE 4th International Conference on cloud computing ,2011,pp. 746-748.

[6] Eun-Kyu Byun and Jin -Soo Kim," Efficient Resource Capacity Estimate of Workflow Applications for Provisioning Resources", USC Technical Report,2008.

[7] E.K. Byun,Y.S. Kee, J. S. Kim, and S. Maeng,"Cost Optimized Provisioning Of Elastic Resources For Application Workflows" Future Generation Computer Systems, Vol.27 , 2011,pp. 1011–1026.

[8] Wang,R.Duan,,X.Li,S.Lu,T.Hung,R.Calheiros, and R.Buyya,"An Iterative Optimization Framework for Adaptive Workflow Management in Computational Clouds",In 11th IEEE International Symposium on Parallel and Distributed Processing with Applications, Los Alamitos, CA, USA, 2013.

[9] G.Lee.,B.G.Chun. and R.H.Katz, "Heterogeneity - Aware Resource Allocation and Scheduling in the Cloud",In 3rd USENIX Workshop on Hot Topics in Cloud Computing Hot- Cloud'11, USENIX Association ,2011.

[10] Daniel de veira, Kary A.C.S Ocalia, Fernanda and Marta,"A Provenance based Adaptive Scheduling Heuristics for Parallel Scientific workflows in clouds",Journal on GridComputing , Vol.10 ,2012,pp. 521 -552.

[11] Rodrigo N. Calheiros and Rajkumar Buyya , "Meeting Deadlines of Scientific Workflows in Public Clouds With Tasks Replication", IEEE Transactions on Parallel and Distributed Systems, 2013, DOI:10.1109/TPDS.2013.238.

[12] Weiwei Chen and Ewa Deelman, "Partitioning And Scheduling Workflows Across Multiple Sites With Storage Constraints",*In 9th international conference on Parallel Processing and Applied Mathematics*,Torun, Poland, 2011.

[13] Murtaza Zafer, Yang Song and Kang-Won Lee,"Optimal Bids for Spot VMs in a Cloud for Deadline Constrained Jobs", *In 11th IEEE conference on Cloud Computing*,2012 ,pp.75-82.

[14] Gurmeet Singh, Mei-Hui Su, Karan Vahi, Ewa Deelman, Bruce Berriman, John Good,Daniel S. Katz, and Gaurang Mehta ,"Workflow Task Clustering for Best Effort Systems with Pegasus*", Proceedings MG'08 of 15th ACM,Mardi Gras Conference,2008.*

[15] Weiwei Chen and Ewa Deelman, "Work flowSim:A Toolkit for Simulating Scientific Workflows in Distributed Environments", *IEEE 8th International Conference on E-Science* ,2012.

[16] N.Rodrigo Calheiros,Rajiv Ranjan, Anton Beloglazov , A.F.CésarDe Rose ,and Rajkumar Buyya,"CloudSim:A Toolkit or Modeling and Simulation of Cloud Computing Environments and Evaluation of Resource Provisioning Algorithms",*Journal Software: Practice and Experience*, Vol. 41,2011 ,pp. 23-50.

[17] http://montage.ipac.caltech.edu.

[18] http://scec.usc.edu/scecpedia/CyberShake .

ANNEXURE

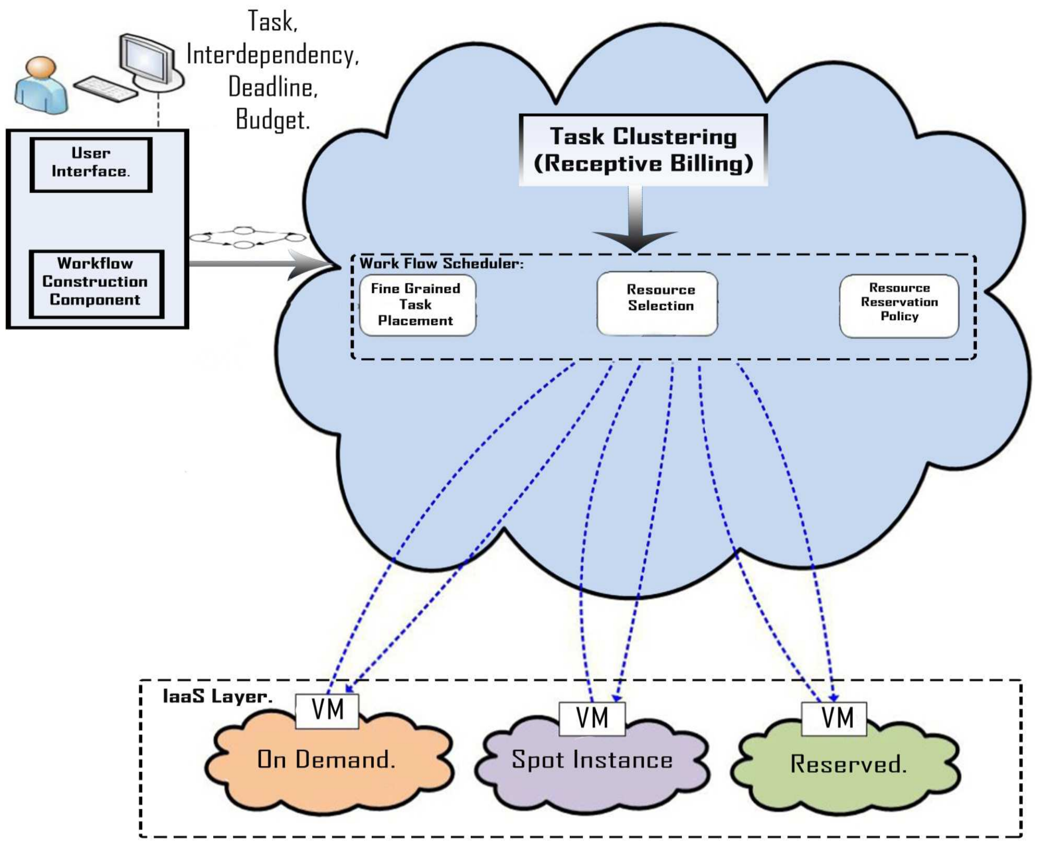
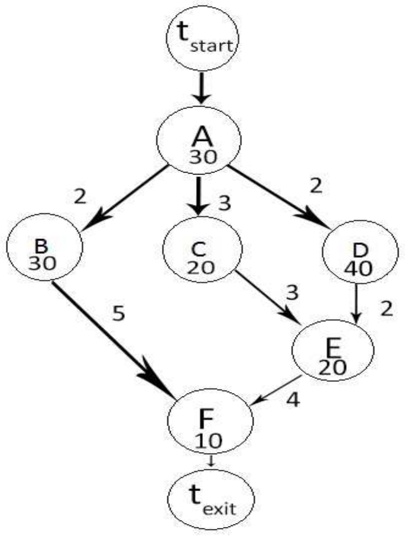


Figure2: Workflow Execution Phases In Cloud Environment



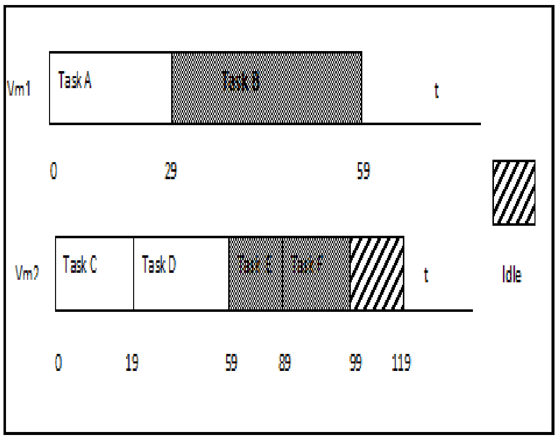


Figure4(A):Example Workflow With Executiontime *Figure 4(C) Proposed Task Clustering.*

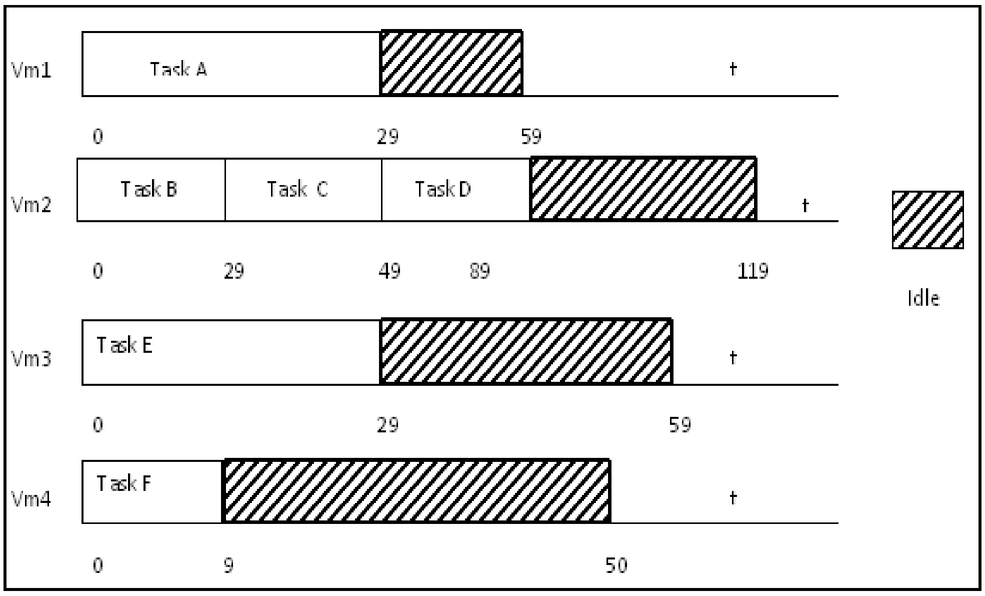


Figure4(B):Workflow Executing Using Level Based Task Clustering.