Australian Journal of Basic and Applied Sciences, 9(6) Special 2015, Pages: 1-4

ISSN:1991-8178

Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com

Monitoring Workflow Applications Execution with Different Resource Types in a Cloud Environment

[1]D.A. Prathibha, [2]Dr. B. Latha, [3]Dr. G. Sumathi

[1]*Research Scholar, Faculty of I &CE, Anna University, Chennai, India*
[2]*Department of CSE, Sri Sairam Engineering College, Chennai, India*
[3]*Department of Information Technology, SVCE, Chennai, India*

ARTICLE INFO

Article history:
Received 12 November 2014
Received in revised form 26 December 2014
Accepted 29 January 2015
Available online 10 February 2015

Keywords:
Cloud Computing, Workflow, Resource Type, Deadline, Budget.

ABSTRACT

Apart from many industrial, business applications being deployed on cloud environment, utilising it for scientific communities for running large scale data and computation intensive applications has evolved. Modelling them as workflow applications raises the main issue of mapping its tasks to an appropriate resource in the cloud environment. Commercial public cloud environments offer different types of resources like Macro, Small, Large and Xlarge which vary in cost and performance. Though the existing systems considers various parameters like execution cost, time and resource utilization for the application Schedulers, they do not consider the unique billing model of the cloud. This work focuses on minimizing the overall execution cost of workflow applications in a given deadline and budget by considering the billing model of the cloud. Performance monitoring of two scientific workflow application execution on different resource types in cloud environment is done.

To Cite This Article: D.A. Prathibha, Dr. B. Latha, Dr. G. Sumathi, Monitoring Workflow Applications Execution with Different Resource types in a Cloud Environment. ***Aust. J. Basic & Appl. Sci.,*** *9(6): 1-4, 2015*

INTRODUCTION

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 [2] have large number of tasks which are interdependent many of which also require parallel execution in order to obtain high performance.

1.1. DAG Workflow Representation:

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} with vertices V = {T1... Tn} denotes the individual task of the workflow and edges E denoting task dependency relationship between the nodes. The DAG also represents the precedence constraints among the tasks i.e. for each (ti, tj) V, tj must be executed after ti completion. Fig.1 shows the DAG representation of workflow, where A, B, F represents a sequence and B, C, D represents parallelism.. 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. Workflow specification containing only task details is called abstract and it is called as concrete, if the graph describes the task as well as where (on which resource) the graph will be executed.

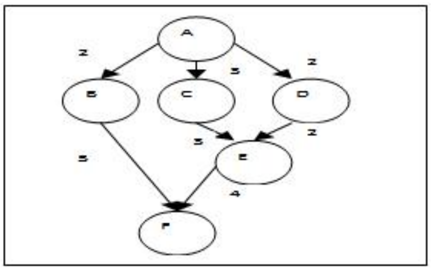

Fig. 1: DAG-based Workflow representation

2.Related work:

Workflow scheduling, a NP-complete problem has lead to proposal of many static heuristics to address the scheduling problem in different platforms (Clusters, Grids). These heuristics try to generate an optimal (or sub-optimal) scheduling plan, i.e. allocate

Corresponding Author: D.A. Prathibha, Research Scholar, Faculty of I & CE, Anna University, Chennai, India.
E-mail: Somaprathi25@gmail.com

activities of a workflow onto a set of available machines prior to the workflow execution.

In (Jia Yu, 2008) Heterogeneous-Earliest-Finish-Time (HEFT) algorithm, execution of each task in all the available resources is calculated by considering dependency of the tasks and a rank that is assigned. Tasks are executed in the descending order of their rank .In [4] the authors propose an extension to HEFT by addressing the elasticity nature of the cloud called Scalable-Heterogeneous-Earliest-Finish-Time in which the resources are dynamically scaled in and out, depending upon the minimum finish time of a given task. In (Boonyarith savapakhiran 2011) the authors propose Simple Super Job scheduling in which jobs are modelled by DAGs and it allows dynamic modelling of DAG that is in this implementation the tasks of the DAG arrive randomly, here the problem of resource underutilization need to be addressed. In this work a set independent or aggregate list of tasks are identified at each level in graph .Afterwards the DAG is divided into multiple levels with each level having set of aggregated tasks. Then scheduling of tasks from the first level to next level happens by considering make span of each level when choosing the resource. In (Daniel de veira, 2012) the authors propose up gradation fit algorithm which based on the make span of the application either does vertical or horizontal optimization. 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 (Long Wang, 2013) author propose progress share algorithm is used for resource allocation in order to have a fair utilization and also introduce job affinity for selecting a resource. Some of the works also consider the repetitive execution feature of the scientific workflow applications for instance in (Daniel de veira, 2012) the authors propose Provenance-based adaptive scheduling heuristic for parallel scientific workflows in cloud which schedules the task based on three factors cost, deadline and reliability. Few works address the issue of identifying minimum number of resources required for given workflow within a given deadline In (Lee, 2011) authors propose Balance time scheduling algorithm to compute minimum number of resources required considering the idle time of the resources in each iteration. Most of the existing work does not consider the dynamic and billing model of the cloud computing for scheduling.

3. Proposed work:

To develop a workflow scheduling technique executes a workflow in a cloud distributed environment which minimizes the SLA violation by taking into consideration budget and deadline. From a catalog of dynamic resources available on public cloud (http://www.aws.amazon.com/ec2/instance-types/), a resource selection mechanism which selects a suitable virtual machine (small, medium and large instance) for a task which checks budget constraint is initially done.

In a public cloud, the resources allocated for minimum period time duration of one hour demands a proposal to combine small tasks and map them on to a single resource (Rodrigo, *et al*). For example if suppose t1 has execution time of 30 min and t2 has execution time 20 minutes and even though they are parallel tasks,t1 and t2 will be mapped to single resource in order utilize fully resource r. Selections made in each level of the DAG are made to meet the deadline and budget constraints.

Develop a set of heuristics that consider the elastic nature of the cloud model in order to decrease the cost of executing workflows on an IaaS cloud provider. The algorithm will dynamically acquire new resources from a cloud platform and will select the best type of resource (instance) to lease based on the characteristics of the task it is intended for. Outline of the proposed system is described below.

Step1: Distribute the Given deadline and budget across each level.

Step2: Identify the order of execution of dependent tasks.

Step 3: Identifying the proper resource in order to get acceptable deadline and budget.

Step 4: For each level of DAG G DO

Find the Cumulative execution time of all the tasks in a given level

Find the cost of execution of all the tasks in a given level

Allocate tasks to the resources based on cost ,execution time and

available time of resource.

Step 5:If Deadline_level[i]<Current _deadline_level[i] then

Modify the resources in the level with higher execution rate.

End if

Step 6:If Budget_level[i]<Current _budget_level[i] then

Modify the resources in the level with low cost available resources.

End if

End loop

4. Experimental Results and Analysis:

Simulation is one of the most popular evaluation methods in scientific workflow studies. Work flow Sim (Weiwei Chen, 2012) provides additional layer for managing workflow in cloud environment. It 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. Workflow applications can be categorized as I/O intensive, Compute Intensive and Storage Intensive based on the tasks in the workflow. Work flow Sim is used in the experiment to monitor the performance of two scientific workflow applications namely Montage and Epigenome.

The first application, Montage (http://www.pegasus.isi.edu), creates science-grade astronomical image mosaics using data collected from telescopes. In our experiments we have considered four cases with number of tasks as 25, 50, 100 and 125. The second application, Epigenome (http://www.pegasus.isi.edu), maps short DNA segments collected using high-throughput gene sequencing machines to a previously constructed reference genome using the MAQ software. The execution of Montage and Epigenome workflows with varied task sets is studied by changing the VM configuration and the underlying scheduling policy.

4.1 Resource Type selection:

This section discusses the experimental evaluation of the impact of resource type on execution of a workflow. The parameters considered are 1) runtime of the work flow, which is defined as the latest finish time on all virtual machines 2) total cost of the workflow which is the sum of product of task durations with the price of the VM allocated to the task. The various instances of virtual machines used in our experiments is given in Table 1.

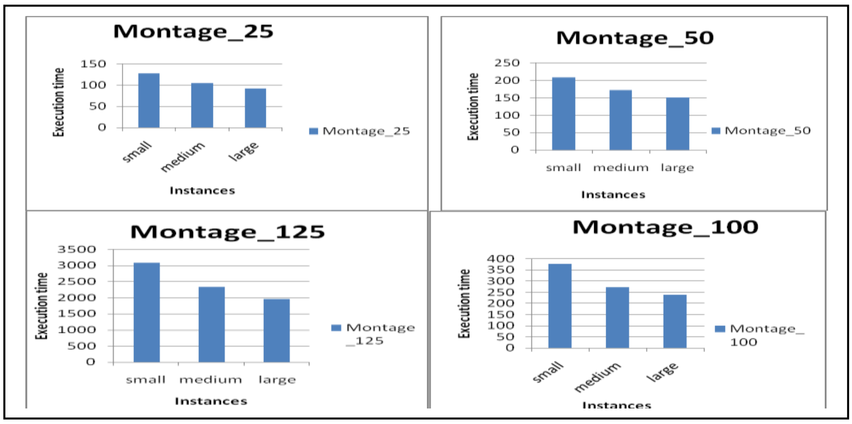

Fig. 2: Execution time of Montage 25, 50, 100 and 125 on different resources.

Table 1: Resource Configuration used in the study experiment.

S.No	Resource Type	RAM (MB)	MIPS	PE
1	Small	2048	1000	2
2	Medium	4096	2000	4
3	Large	8192	4000	8

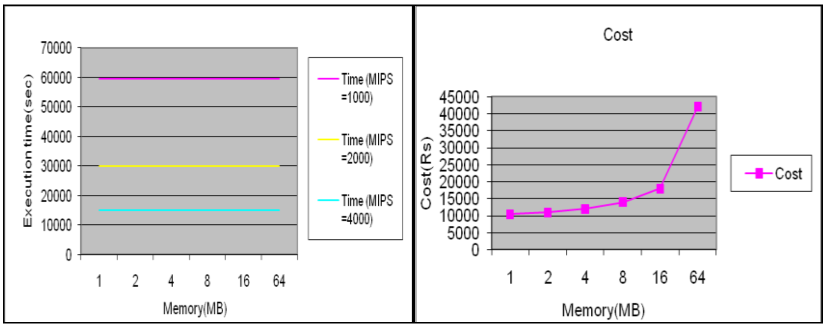

Fig. 3: Execution time and cost in min-min scheduling.

From the Figure (Jia Yu, 2008) it can be observed that irrespective of the workflow size, higher resource capacity yields shorter make span.

4.2 Job Affinity and Scheduling Heurisitics:

This section details the relationship between job affinity and the resource type. Initially the workflow was executed by varying the memory capacity keeping the processing speed as constant. Next the same workflows were executed with varied processing speed and constant memory. It is observed that increased processing speed, results in reduced execution time irrespective of the memory size. In this study analysis of relationship between job and the resource is carried out. Next the workflow was executed with different values of processing speed and memory size was kept constant in this case it was observed that execution time was reduced significantly result shows that there is a increase in the cost value and there is no change in the execution time since Epigenome workflow application is compute intensive. Fig 3 shows the result of executing Epigenome with 100 tasks on different virtual machines.

The experiment was conducted with Min-Min (Jia Yu, 2008) as the underlying scheduling policy was implemented in which task with minimum execution time is executed on the resource which takes minimum time.

The experiment was repeated with HEFT (Jia Yu, 2008) scheduling policy in which scheduling_is done by assigning ranking value to each task on a given resource.

Table 2: Epigenom_100 workflow on different types of resources with HEFT scheduling.

Memory (MB)	PE	Execution Time(seconds)			Cost
		MIPS =1000	MIPS =2000	MIPS =4000	
1048	2	56695	26302.21	13996.95	10524
2048	2	55411.81	27081.6	13227.89	11024
4048	2	56352.7	27390.43	13977.4	12024
8048	2	53907.82	27153.65	13526.25	14024
16048	2	53599.79	26851.55	13537.69	18024
64048	2	55284.96	26581.82	13574.34	42024

Table 2 shows workflow executes faster on resource with MIPS value of 4000,but even there is no significant change in the execution time for different memory size

Conclusion:

Cloud computing is making its presence in various domains for its cost effective and energy saving benefits. In this paper we have proposed workflow scheduling considering the heterogeneous nature of the cloud and billing model of the cloud is also considered in the resource selection phase of the workflow scheduling. Experiments were conducted using WorkflowSim for two workflows Montage and Epigenome, on different types of resource(small,medium and large). From the results it is clear that resource size and type has the significant impact on the performance of workflow applications along with the scheduling heuristics.

REFERENCES

Boonyarith savapakhiran, George Michailidis, Michael Devetsikiotis, 2011. Aggregated-DAG scheduling for job flow maximization in heterogeneous cloud computing, *In Proc IEEE Globecom* 2011.

Cui Lin and Shiyong Lu, 2011. Scheduling Scientific Workflows Elastically for cloud computing , *In Proc IEEE international conference on cloud computing*, pp: 746-748.

Daniel de veira, Kary A.C.S Ocalia, Fernanda and Marta, 2012. A Provenance Adaptive Scheduling Heuristics for Parallel Scientific workflows in clouds., *J.GridComputing*, pp: 521-552.

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

http://www.aws.amazon.com/ec2/instance-types/

http://www.pegasus.isi.edu

Jia Yu,Rajkumar Buyya and Kitagiri Rammohanarao, 2008. Workflow scheduling algorithms for Grid Computing, Chapter 5 *Studies in Computational Intelligence*, 146: 173-214.

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

Long Wang, Rubing Duan, Li. Xiaorong, Lu. Sifei, Terence Hung, Rodrigo Calheiros And Rajkumar Buyya, 2013. An Iterative Optimization Framework for Adaptive Workflow Management in Computational Clouds, *In Proc IEEE conference*.

Parallel and Distributed Systems, (in press).

Rodrigo, N., Calheiros & Rajkumar Buyya, Meeting Deadlines of Scientific

Weiwei Chen, Ewa Deelman, 2012. Work flow Sim: A Toolkit for Simulating Scientific Workflows in Distributed Environments, *In Proc.Of E-Science, IEEE 8 th International Conference*, pp: 1-8.

Workflows in Public Clouds with Tasks Replication, *IEEE Transactions on*