An analysis of Adaptive Heuristic Scheduling for Multiuser Cloud

Rashmi S
Department of Computer Science and Engineering,
East Point College of Engineering and Technology,
Bengaluru, India

Dr. Anirban Basu
Department of Computer Science and Engineering
APS College College of Engineering, Bengaluru,
India

***Abstract*- Cloud computing has significant features such as on-demand delivery, elasticity and multitenancy. Virtualization is a dominant technology that powers cloud computing. A Hadoop- MapReduce cluster can be shared to enable multiuser to execute parallel jobs. A Hadoop datacenter consists of a large number of physical servers that lets that cloud providers meet the requirements of the users. A Balanced and efficient usage of these servers is required to improve the utilization and to reduce execution time of the jobs. A Provenance database can be used to track the root and path of the server usage. In this paper, we analyze an Adaptive Heuristic Scheduling method which makes use of provenance data to complete all jobs within their deadlines with reduced cost.**

Keywords - Cloud computing, adaptive heuristic scheduling, virtualization, provenance, deadline

I. INTRODUTION

Cloud computing [1] known for its on-demand computing, is a kind of Internet-based computing that provides shared processing resources and data to computers and other devices on demand. Elasticity is one of the important characteristics of cloud computing which allows scaling out and quickly scaling in the services. Multitenancy is yet another characteristics where the pooled resources serve multiuser dynamically as per their demands.

In the recent times, MapReduce [2] provides a cost-effective solution for running parallel applications. It parallelizes across hundreds and thousands of CPUs automatically. MapReduce runs multiple maps and reduce tasks and parallelize the computations.

Virtualization [3] is fundamental to cloud computing. Cloud providers have large data centers with huge number of servers to power their cloud services. Virtualization allows us to virtually partition the data on the server, enabling each client to work with a separate “virtual” instance of the same software. Virtual machines automatically instantiate and terminate the execution of job. Virtual Machines are requested on demand from the Hadoop data centre infrastructure. The VMs are allocated to servers. The servers may be idle or may be running other VMs parallely. The concurrently executing VMs may be utilizing the resources heavily which could deteriorate the performance of other jobs in a multiuser environment. Hence the need for a proficient scheduling mechanism, to balance the load on different servers.

Provenance [4][5][6] refers to information that gives a record of data product, starting from its root. This information is beneficial in data-intensive computing scenarios, such as scientific computing, to ensure QOS. In cloud computing environment, a provenance system is used to gather and pile up related metadata. Later, the stored metadata is used for verification and tracking back, fault detection assurance of reproducibility, security and audit trials [5]. In Adaptive Heuristic [3], scheduling VMs on the server, takes advantage of this trait of a provenance system.

II. LITERATURE SURVEY

Round Robin Scheduling algorithm is the simplest scheduling technique. It divides time into multiple time quantum. Each waiting job is given the same execution time If a job is unable to complete in the given quantum, it has to wait until it gets back after it completes a ring.

In [7] M Dabbagh et al have proposed an energy-aware resource provisioning framework. Energy efficiency is a key concern of this framework due to major issue in large data centers in business environments It believes in foreseeing the total number of possible VM requests that may come up and the quantity of upcoming resources from these requests. Based on these predictions, the appropriate number of Physical machines (PM) required are decided. The factors need to be considered are length of the observation window, , the length of the prediction window, the optimal weights of the stochastic predictor and number of clustered categories. The experimental evaluations show that the energy-aware resource provisioning framework makes ample energy savings.

In [8], the challenges in collecting provenance for cloud computing are described. In recent days, Data Provenance in cloud has gained more popularity. Several applications are now dependent on in cloud infrastructures. Data provenance provides historical data from its original resources and builds reliable relations between cloud providers and cloud users. But the entire burden of maintaining and updating provenance is on the administrators. Mohd Izuan Mohd Saad et al in [8] have presented a model to securely access provenance data in a using a secured communication channel.

In Cloud computing environments, data is growing at an enormous rate. The power to track and trace its footprints is of major concern. Chun Hui Suen et al, raises the need for two important factors. i) The need for data tracking tools. ii) The need for system –centric as well as data- centric logging mechanism to track the activities in cloud .The activities refers to file operations such as file creation, edition, duplication, transfers, deletions, etc. within and across all cloud servers. In [9], S2Logger, a data event logging mechanism which captures analyses and visualizes data events in the cloud from the data point of view is introduced. S2Logger monitors data provenance as a graph. It works at both file –level and block-level. Data logs are analyzed based on graph representations. It also takes care of data leakages at control points. It also addresses data-related cloud security problems such as data policy violations malicious actions, data policy violations and data policy violations by analyzing the data provenance. S2Logger also enables us to address the gaps and inadequacies of existing system-centric security tools.

III. ADAPTIVE HEURISTIC SCHEDULING METHOD

In Adaptive Heuristic Scheduling method [3][10], the servers are allotted with multiuser parallel jobs. The algorithm makes use of a flag to balance the server utilization. It also employs a provenance database to track the root and path of server usage. This assists the scheduling method to choose a low usage server. The design is shown in Fig 1.

Cloud manager: Cloud manager is responsible for management of cloud in order to ensure that all cloud computing based resources are working optimally and properly with user and other services.

Request processor: Request processor takes the input in the form of workflow and deadline. The workflow is represented using a Directed Acyclic Graph (DAG).Fig 2 shows an example workflow in the form of a DAG. Job1 is the first job to be executed in the set of jobs = {1, 2, 3, 4, 5, 6}.Jobs 2 and 3 can be executed in parallel but after Job1.Jobs 4 and 5 are dependent on jobs 2 and 3. Job6 should be the last job to be executed. It processes the request and forwards the request to task dead line calculator.

Task deadline calculator: The user will set the deadline among the various tasks before submitting the task to the VM. The task dead line calculator is use full for resource provisioning. Fig shows the sequence diagram for running a task on VM Flow

Resource provisioning: Resource provisioning means the selection, deployment, and run-time management of software and hardware resources for ensuring guaranteed performance for applications.

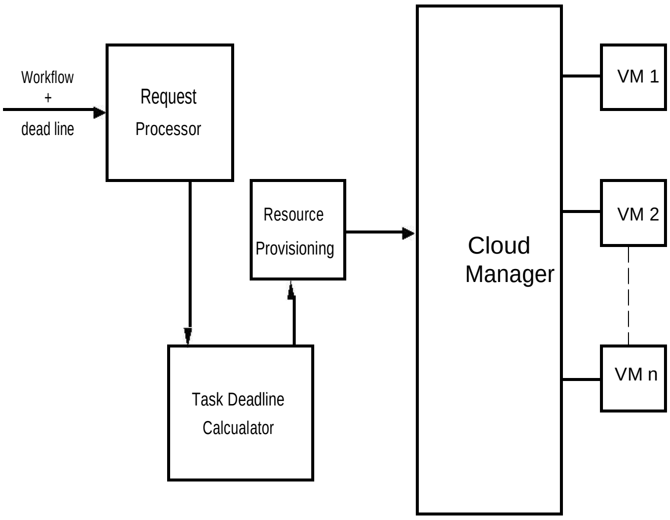


Fig 1: Design of provenance based Scheduling [9]

The algorithm balances the server usage using a timer and flag. The server is said to be active if it has sufficient resources to execute a job. Otherwise it is treated as an idle server. Initially, the timer is preset to a constant value for each individual server and the flag is set to 1. The timer starts decrementing once a job is assigned to the server. Subsequently, when the timer reaches "0", its flag is reset. The server is treated as an idle server. When the server is set back to "1", it becomes an active server.

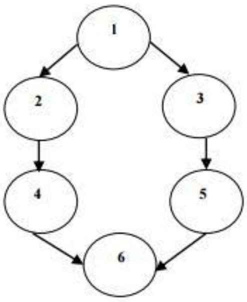


Fig 2: DAG representation of a workflow

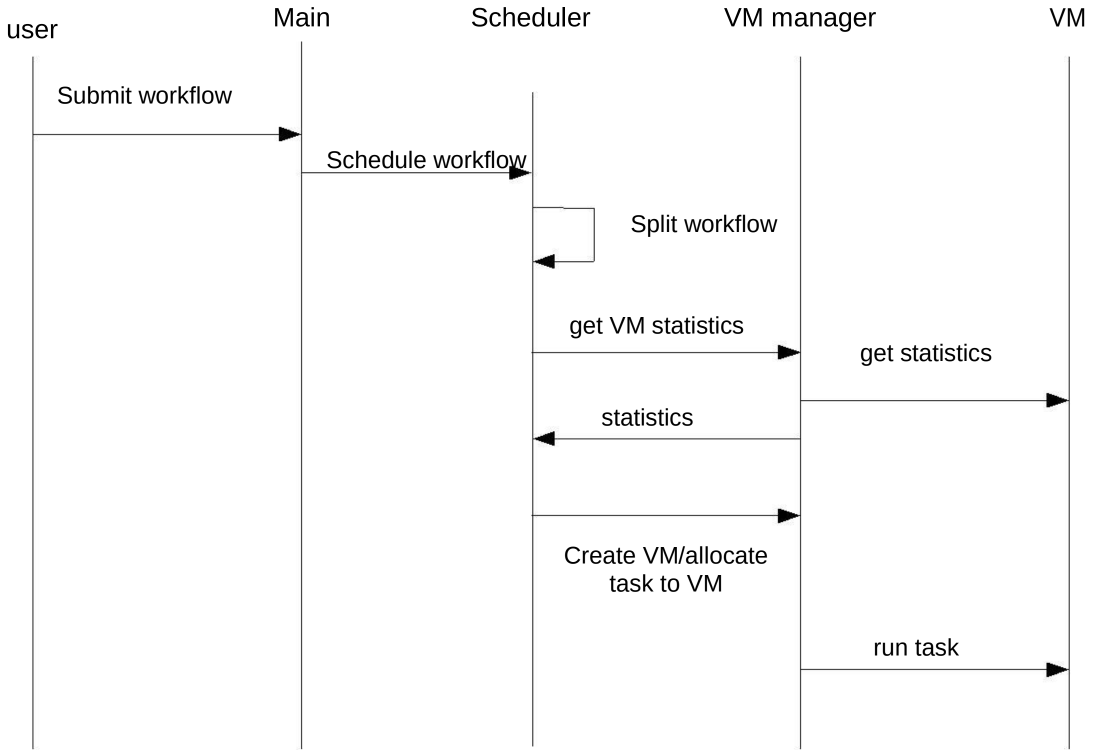


Fig 3: Sequence diagram for running a task on VM Flow

The algorithm [10] for Adaptive Heuristic Scheduling is shown below:

Start

Initialize datacenter and time slot and machine size Schedule the each user jobs as follows

If flag = 1
Server is
active

Can allocate job to this
server else

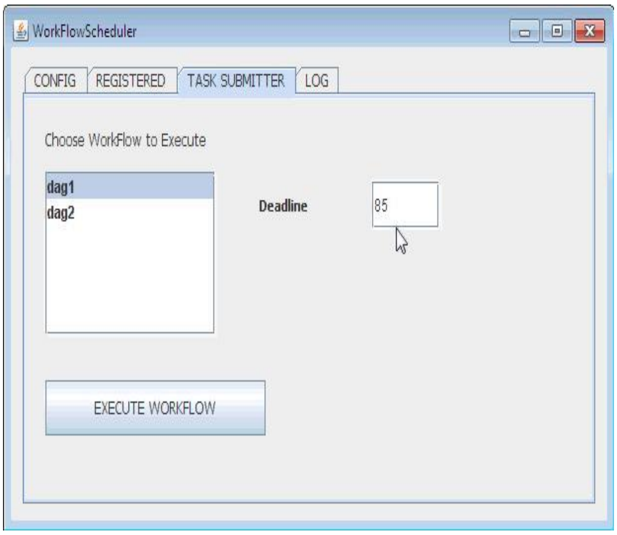
Flag = 0
Server is
Idle

Choose another server

Update the provenance
database End

IV. EXPERIMENTAL RESULTS

The experiment was conducted on Intel Core 2 Duo 2.10 GHz processor and 3 GB of RAM, Ubuntu 9.10 platform and using CloudSim simulator.
Fig 4 shows workflow given as a DAG and deadline specified.



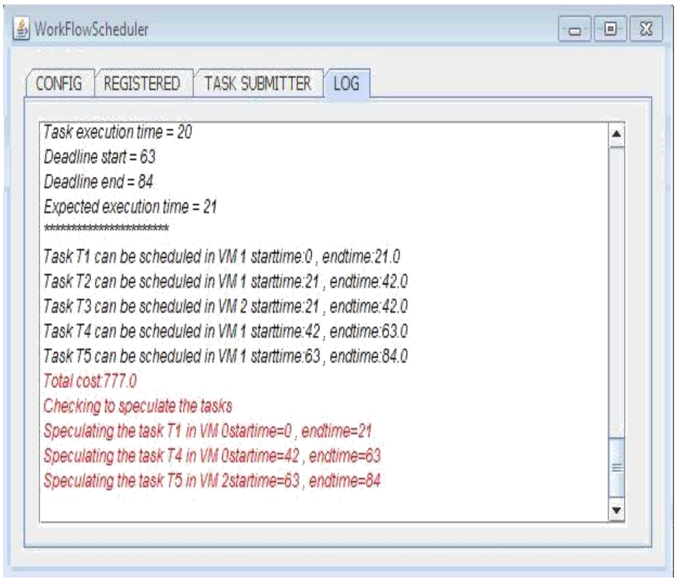
Choose WorkFlow to Execute

dag1
dag2

Deadline 85

Fig 4: Workflow and Deadline specified

Fig 5 shows the tasks scheduled on different VMs and tasks completed before deadline.



```
Task execution time = 20
Deadline start = 63
Deadline end = 84
Expected execution time = 21
************************
Task T1 can be scheduled in VM 1 starttime:0 , endtime:21.0
Task T2 can be scheduled in VM 1 starttime:21 , endtime:42.0
Task T3 can be scheduled in VM 2 starttime:21 , endtime:42.0
Task T4 can be scheduled in VM 1 starttime:42 , endtime:63.0
Task T5 can be scheduled in VM 1 starttime:63 , endtime:84.0
Total cost:777.0
Checking to speculate the tasks
Speculating the task T1 in VM 0starttime=0 , endtime=21
Speculating the task T4 in VM 0startime=42 , endtime=63
Speculating the task T5 in VM 2startime=63 , endtime=84
```

Fig 5 : VMs scheduled and executed

The results are shown graphically. A comparative study of Adaptive Heuristic Scheduling based on Provenance allocation is made with Round Robin Scheduling.

Fig 6 shows the comparison based on cost and Fig 7 shows the comparison based on Deadline meet ratio.

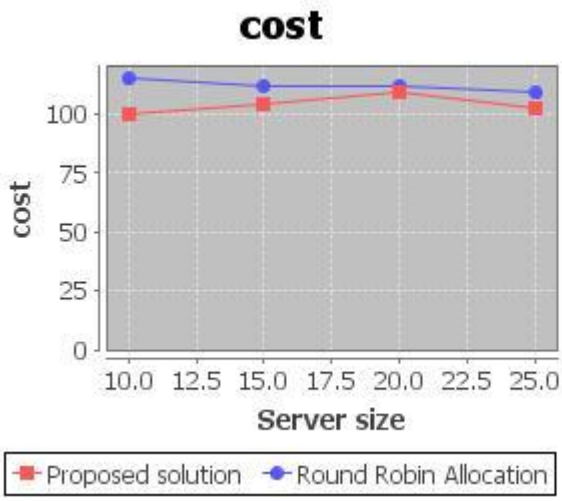


Fig 6: Cost based comparison of Provenance allocation and Round Robin Allocation

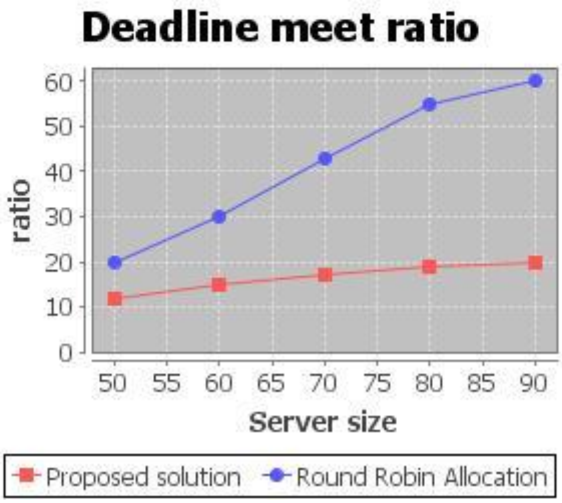


Fig 7: Deadline meet ratio based comparison of Provenance allocation and Round Robin Allocation

V. CONCLUSION

This paper analyses the adaptive heuristic scheduling which provides an effective solution for allocating parallel jobs. The scheduling method uses a provenance database to allocate multiuser jobs without collision. A flag is used to differentiate between an idle and active server. An active server has enough resources to handle jobs whereas an idle server doesn't. The provenance database is updated after each allocation .The experimental result shows the reduction of cost and a improved deadline meet ratio.

REFERENCES

[1] NIST Definition of Cloud Computing v15, csrc.nist. Gov/groups/SNS/cloud-computing/cloud-def-v15 .doc

[2] J Dean and Ghemawat,"MapReduce: Simplied Data Processing on Large Clusters", Google, Inc., 2004

[3] Daniel de Oliveira , Kary A. C. S. Ocaña , Fernanda Baião , Marta Mattoso," A Provenance-based Adaptive Scheduling Heuristic for Parallel Scientific Workflows in Clouds", Springer Science+Business Media B.V. 2012

[4] Muhammad Imran, Helmut Hlavacs, "Provenance in the Cloud: Why and How?", The Third International Conference On Cloud Computing, Grids, And Virtualization, CLOUD COMPUTING 2012

[5] I. M. Abbadi and J. Lyle, "Challenges for provenance in cloud computing," in Proc of the 3rd USENIX Workshop on the Theory and Practice of Provenance. USENIX, 2011.

[6] Daniel Crawl, Jianwu Wang, Ilkay Altintas," Provenance for MapReduce-based Data-Intensive Workflows", *WORKS'11,* November 14, 2011, Seattle, Washington, USA.

[7] M. Dabbagh, B. Hamdaoui, M. Guizani and A. Rayes, "Energy-efficient resource allocation and provisioning framework for cloud data centers", Network and Service Management, IEEE Transactions on, no. 99, pp. 1-1, 2015

[8] Mohd Izuan Mohd Saa and Kamarularifin Abd Jalil, "Data provenance trusted model in cloud computing", International Conference on Research and Innovation in Information Systems (ICRIIS'13), 2013.

[9] C. H. Suen, R. K. L. Ko, Y. S. Tan, P. Jagadpramana and B. S. Lee, "S2Logger: End-to-End Data Tracking Mechanism for Cloud Data Provenance", *IEEE Press*, pp. 594-602

[10] I.M.Maywish Rajakumari, Mrs.R.Narayani," , Provenance based Adaptive Heuristic Scheduling in Cloud Environment", International Journal of Scientific & Engineering Research, Volume 6, Issue 4, April-2015