A SURVEY OF EXISTING WORKFLOW SCHEDULING ALGORITHMS IN CLOUD COMPUTING

Jithin V S*

G. Kharmega Sundararaj**

I. Abstract

In cloud computing, computing resources are provided on demand. Input to the cloud systems include workflows with different constrains. Scheduling of workflows determines the performance of the cloud systems. Since cloud computing extends Grid computing and Virtualization concepts scheduling algorithms used in these technologies can also be used in cloud computing. This paper presents a survey of existing scheduling algorithms and helps to chose appropriate scheduling algorithm for a workflow with various constrains.

Key words- Cloud Computing, Workflow Scheduling Algorithms, Workflow

* M.E Student, PSN College of Engineering & Technology, Tirunelveli, India

** PSN College of Engineering & Technology, Tirunelveli, India

II. Introduction

Cloud computing is an extension of parallel computing, distributed computing and grid computing. Now days most software and hardware have provided support to virtualization. Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction[1]. This cloud model is composed of five essential characteristics**,** three service models, and four deployment models. Essential Characteristics of cloud computing are On-demand self-service, Broad network access, Resource pooling, Rapid elasticity, Measured service.

The primary benefit of moving to Clouds is application scalability. Unlike Grids, scalability of Cloud resources allows real-time provisioning of resources to meet application requirements. Cloud services like compute, bandwidth resources are available at substantially lower costs. Usually tasks are scheduled by user requirements. New scheduling strategies need to be proposed to overcome the problems posed by network properties between user and resources. New scheduling strategies may use some of the conventional scheduling concepts to merge them together with some network aware strategies to provide solutions for better and more efficient job scheduling. Usually tasks are scheduled by user requirements.

Initially, scheduling algorithms were being implemented in grids. Due to the reduced performance faced in grids, now there is a need to implement scheduling in cloud. The primary benefit of moving to Clouds is application scalability. Unlike Grids, scalability of Cloud resources allows real-time provisioning of resources to meet application requirements. This enables workflow management systems to readily meet Quality of- Service (QoS) requirements of applications, as opposed to the traditional approach that required advance reservation of resources in global multi-user Grid environments. Cloud services like compute, storage and bandwidth resources are available at substantially lower costs. Cloud applications often require very complex execution environments .These environments are difficult to create on grid resources. In addition, each grid site has a different configuration, which results in extra effort each time an application needs to be ported to a new site. Virtual machines allow the application developer to create a fully customized, portable execution environment configured specifically for their application.

Traditional way for scheduling in cloud computing tended to use the direct tasks of users as the overhead application base. The problem is that there may be no relationship between the overhead application base and the way that different tasks cause overhead costs of resources in cloud systems. For large number of simple tasks this increases the cost and the cost is decreased if we have small number of complex tasks.

A workflow models a process as consisting of a series of steps that simplifies the complexity of execution and management of applications [2]. Scientific workflows in domains such as high-energy physics and life sciences utilize distributed resources in order to access, manage and process large amount of data from a higher-level. Processing and managing such large amounts of data require the use of a distributed collection of computation and storage facilities. There is many kind of workflows with various constrains they are Data intensive workflows, Instance intensive workflows, Time constrained workflows. Workflow scheduling is an NP-hard problem.

III. Scheduling Problems for Cloud Computing

On the contrary a problem is in Class NP-complete if its purpose is making a decision, and is in Class NP-hard if its purpose is optimization. Since cloud workflow scheduling need resource optimization it comes under NP-hard class. Because an optimization problem is not easier than a decision problem, we only list schematic methods for NP-hard problems. Enumeration, heuristic and approximation are three possible solutions; their corresponding algorithms complement each other to give a relatively good answer to a NP-hard problem.

IV. Existing Workflow Scheduling Algorithm in Clouds

Cloud computing is a distributed computing technology which extends existing technologies such as Grid computing, Virtualization etc. Algorithms used in these technologies can also be used in cloud computing. Following Scheduling Algorithms are currently used in clouds.

4.1 Heterogeneous Earliest- Finish-Time Algorithm

HEFT algorithm is proposed by Tannenbaum et al in 2002 and used by ASKALON (Fahringer et al., 2005)[5]. Scheduling algorithm was designed for heterogeneous computing systems. Since it was developed before the advent of cloud computing and utility grids, it does not consider monetary costs. Its objective is to minimize the workflow makespan.

4.2 Deadline-driven cost-minimization Algorithm

Deadline-Markov Decision Process (MDP)[6] breaks the DAG into partitions, assigning a maximum finishing time for each partition according to the deadline set by the user. Based on this time, each partition is scheduled for that resource, which will result in the lowest cost and earliest estimated finishing time. This algorithm works with on-demand resource reservation.

4.3 Partial Critical Paths Algorithm

Abrishami *et al.* [7] presented the Partial Critical Paths (PCP) algorithm, which schedules the workflow in a backward fashion. Constraints are added to the scheduling process when such scheduling of jobs in a partial critical path fails so that the algorithm will be restarted. This algorithm presents the same characteristics as does MDP, although it involves greater time complexity, since a relatively large number of rescheduling can be demanded during the execution of the algorithm.

4.4 The Hybrid Cloud Optimized Cost

The Hybrid cloud optimized cost algorithm [3] schedules workflows in hybrid clouds by first attempting costless local scheduling using HEFT. If the local scheduling cannot meet the deadline, the algorithm selects jobs for scheduling in resources from the public cloud. When selecting resources from the public cloud, the HCOC algorithm considers the relation between the number of parallel jobs being scheduled and the number of cores of each resource as well as deadlines, performance, and cost. As with the MDP algorithm, the objective is to minimize the financial cost, obeying the deadlines stipulated by the user in a single-level SLA contract.

4.5 Min-Min Heuristic Algorithm

Min-min algorithm is proposed by Maheswaran et al in 1999[4]. Min-min firstly updates the set of arrival tasks and the set of available machines, calculating the corresponding expected completion time for all ready tasks. Next, the task with the minimum earliest completion time is scheduled and then removed from the task set. Machine available time is updated, and the procedure continues until all tasks are scheduled.

4.6 Max-Min Heuristic Algorithm

Min-min algorithm is proposed by Maheswaran et al. Max-min heuristic [4] differs from the Min-min heuristic where the task with the maximum earliest completion time is determined and then assigned to the corresponding machine. The Max-min performs better than the Min-min heuristic if the number of shorter tasks is larger than that of longer tasks.

4.7 Balanced Time Scheduling Algorithm

Balanced Time Scheduling (BTS) heuristic algorithm is by proposed Byun [12].BTS estimates the minimum number of computing resources required to execute a workflow within a given deadline. BTS is cost-efficient, scalable, and generic. The resource estimate of BTS is abstract, so it can be easily integrated with any resource description language or any resource provisioning system.

4.8 A Particle Swarm Optimization-based Heuristic Algorithm

A Particle Swarm Optimization-based Heuristic for Scheduling Workflow Applications: Suraj Pandey, LinlinWu, Siddeswara Mayura Guru, Rajkumar Buyya [13] presented a particle swarm optimization (PSO) based heuristic to schedule applications to cloud resources that takes into account both computation cost and data transmission cost. It is used for workflow application by varying its computation and communication costs. The experimental result shows that PSO can achieve cost savings and good distribution of workload onto resources.

4.9 Compromised-Time-Cost Scheduling Algorithm

Compromised-time-cost (CTC) scheduling algorithm [8] considers the characteristics of cloud computing to accommodate instance-intensive cost-constrained workflows by compromising execution time and cost with user input enabled on the fly. CTC Scheduling Algorithm is used in SwinDeW-C for Instance-Intensive Cost-Constrained Workflows on a Cloud Computing Platform

4.10 Max-Min Fair Scheduling Algorithm

Max-Min Fair Share (MMFS)[9] scheduling policy is developed for computational Grids which simultaneously address the problem of finding a fair task order and assigning a processor to each task based on a max-min fair sharing policy. MMFS policy reduces resource provisioning cost by increasing resource utilization.

4.11 Optimized Resource Scheduling Algorithm

Hai Zhong, Kun Tao, Xuejie Zhang proposed an optimized resource scheduling algorithm that focuses on achieving the optimization or partial optimization for cloud scheduling outperforms [10]. This algorithm uses an improved genetic algorithm for automation scheduling policy to improve utilization rate of resources and speed.

4.12 Improved Cost-Based Algorithm

Y. Yang, K. Liu, J. Chen, X. Liu, D. Yuan and H. Jin introduce an improved cost-based algorithm to improve the computation / communication rate in cloud computing [11]. This algorithm makes an efficient mapping of tasks through measurement of resource cost and computation performance.

Table 1 lists various workflow scheduling algorithms and target system and applicability for cloud scheduling. Although not all scheduling algorithms used in clouds were conceived for these systems.

Table 1

Scheduling Algorithm	Target System	Optimization criteria	Suitable Workflow
HEFT[5]	Heterogeneous	minimize makespan	Instance-Intensive
MDP [6]	Utility Grid	minimize cost within deadline	Time-Constrained
PCP [7]	Utility Grid	minimize cost within deadline	Time-Constrained
HCOC [3]	Cloud	minimize cost within deadline	Time-Constrained
Min-Min[4]	Distributed, Cloud	minimize makespan	Instance-Intensive
Max-min[4]	Distributed	minimize makespan	Instance-Intensive
PSO based Heuristics[13]	Cloud	minimize cost	QOS-Constrained workflow
BTS[12]	Grid	Minimize resource	Time-Constrained
CTC[8]	Cloud	Comprise time and cost	instance-intensive cost-constrained
MMFS[9]	Utility Grid	minimize cost within deadline	Time-Constrained
Optimized Resource Scheduling [10]	Cloud	Improve resource utilization	Instance-Intensive

Improved Cost Based [11]	Cloud	minimize cost	Transaction-Intensive Cost-Constrained

V. Conclusion

In cloud computing workflow scheduling determines the performances of the cloud system. A workflow should be scheduled with most appropriate algorithm based on its characteristics. The algorithms used in Grid computing and Virtualization can also be used in cloud systems. In this paper the existing workflow scheduling algorithms that can be used in cloud systems are analyzed and workflows suitable for each algorithm are found.

VI. Reference

[1] "The NIST Definition of Cloud Computing" Special Publication 800-145

[2] Wiley STM / Editor Buyya, Broberg, Goscinski: "*Cloud Computing: Principles and Paradigms"*

[3] L. F. Bittencourt and E. R. M. Madeira, "HCOC: A Cost Optimization Algorithm for Workflow Scheduling in Hybrid Clouds," *J. Internet Svcs. and Apps.*, vol. 2, no.3, Dec 2011, pp. 207–27.

[4] Braun, T.D., H. Jay Siegel, N. Beck, L.L. Boloni, M. Maheswaran, A.I. Reuther, J.P. Robertson, M.D. Theys and B. Yao, 2001. A Comparison of Eleven Static Heuristics for Mapping a Class of Independent Tasks onto Heterogeneous Distributed Computing Systems. Journal of Parallel and Distributed Computing, 61: 810-837.

[5] H. Topcuoglu, S. Hariri, and M.-Y. Wu, "Performance-Effective and Low-Complexity Task Scheduling for Heterogeneous Computing," *IEEE Trans. Parallel and Distrib. Sys.*, vol. 13, no. 3, 2002, pp. 260–74.

[6] J. Yu, R. Buyya, and C. K. Tham, "Cost-based Scheduling of Scientific Workflow Applications on Utility Grids," *Int'l. Conf. e-Science and Grid Computing*, July 2005, pp. 140–47.

[7] S. Abrishami, M. Naghibzadeh, and D. Epema, "Cost-Driven Scheduling of Grid Workflows Using Partial Critical Paths," *11th IEEE/ACM GRID*, Oct. 2010, pp. 81–88.

[8] K. Liu; Y. Yang; J. Chen, X. Liu; D. Yuan; H. Jin, A Compromised-Time- Cost

Scheduling Algorithm in SwinDeW-C for Instance-intensive Cost-Constrained Workflows on Cloud Computing Platform, International Journal of High Performance Computing Applications, vol.24 no.4 445-456,May,2010.

[9] “Fair Scheduling Algorithms in Grids” IEEE Transactions On Parallel And Distributed Systems, Vol. 18, No. 11, November 2007

[10] Hai Zhong, Kun Tao, Xuejie Zhang, "An Approach to Optimized Resource Scheduling Algorithm for Open-source Cloud Systems “, 5th Annu. Conf. China Grid Conference, China, 2010

[11] Y. Yang, K. Liu, J. Chen, X. Liu, D. Yuan and H. Jin, "An Algorithm in SwinDeW-C for Scheduling Transaction-Intensive Cost-Constrained Cloud Workflows", 4th IEEE International Conference on e-Science, 374-375, Indianapolis, USA, December 2008.

[12] E.-K. Byun, Y.-S. Kee, E. Deelman, K. Vahi, G. Mehta, J.-S. Kim, Estimating resource needs for time-constrained workflows, in: Proceedings of the 4th IEEE International Conference on e-Science, 2008

[13] Suraj Pandey1; LinlinWu1; Siddeswara Mayura Guru; Rajkumar Buyya, A Particle Swarm Optimization-based Heuristic for Scheduling Workflow Applications in Cloud Computing Environments.