A SURVEY ON VARIOUS TASK SCHEDULING ALGORITHM IN CLOUD COMPUTING

Gaurang Patel[1], Asst.Prof. Rutvik Mehta[2]
[1] *PG Student, Computer Science & Engineering Department, PIET, Vadodara.*
[2] *Assistant Professor, Information Technology Department, PIET, Vadodara.*

Abstract— **For today's most demanding service of cloud computing, there are many tasks requires to be executed by the available resources to achieve best performance, shortest response time and utilization of resources. To achieve these challenges there is a need to design a new task scheduling algorithm that outperform appropriate allocation map of task. In this paper, a systematic review of various task scheduling algorithm is presented. These algorithms have different perspective, working principles etc. This study concludes that all the existing techniques mainly focus on reducing service response time, minimize completion time, maximize utilization of resources and improving performance etc. There are many parameters can be mentioned as factor of scheduling problem to be considered such as load balancing, system throughput, service reliability, service cost, service utilization and so forth.**

Index Terms— **Cloud Computing, Meta-Tasks, Makespan, Response Time and Min-Min algorithm.**

I. INTRODUCTION

Cloud computing provide a digital service delivery over the internet by various applications that are accomplished by computer systems in distributed data centers [1][3]. Cloud Computing is getting more advanced now days. Cloud service providers are aimed to provide services using large scale cloud environment with cost effectiveness. Also, there are few popular large scaled applications like social-networking-commerce etc. These applications can benefit to minimize the costs using cloud computing. It provides infrastructure, platform, and software which are made available as subscription-based services in a pay-as-you-go model to consumers. Those services are known as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) in industries[4].Cloud computing is known as internet based computing service provided by various infrastructure providers on an on-demand basis, so that cloud is subject to Quality of Service (QoS), Load Balance (LB) and other constraints which have direct effect on user consumption of resources controlled by cloud infrastructure.

As the number of cloud users are increasing rapidly, assuring the optimum use of cloud resources are difficult task for cloud service providers and there are many issues associated with the task scheduling algorithms and so the load balancing for static meta-task described in can be used for cloud for balancing the load[11]. The objective here is to Study about a variety of load balancing algorithms for cloud system and to invent and develop the concept of load balancing and maximum makespan using the data of different cloud servers. Thus it evaluates the performance of the system. The set of assumptions here are that tasks have no deadlines and priorities should be assigned. The mapping Process is to be executed statically in a batch mode fashion. The size of meta-task should be well known in advance. Security and others issues are assumed to be satisfied.

The work here is limited to set of assumption. This can be implemented as a separate in itself. This paper only focuses on the issue of load balancing and poor makespan. Tasks have no deadlines or priorities assigned. The Mapping Process is to be executed statically in a batch mode fashion. The size of meta-task should be well known in advance. Security and others issues are assumed to be satisfied. This research cloud is adapted to a broader variation of combinational resource allocation/scheduling problem.

A. Task Scheduling in Cloud Computing

In Cloud, So many tasks have to be executed by the available resources to achieve distinct intensions. There is a need for a scheduling algorithm that is used by task scheduler to outperform appropriate allocation map of tasks on resources.Distinct types of scheduling are based on different criteria, such as Static or Dynamic scheduling, Centralized/Distributed scheduling, Pre-emptive or Non-Preemptive scheduling, Cooperative scheduling, Immediate or Online mode scheduling, Batch or Offline mode scheduling etc.

B. Orientation of paper

Section 1 Introduction of cloud computing is explained briefly. In Section 2, it contains task scheduling techniques & problems in existing algorithms, In Section 3, Comparison based on different factor is mentioned and finally in Section 4specifies conclusion.

II. VARIOUS TASK SCHEDULING TECHNIQUE

There are different task scheduling algorithms is used in the context of different factors in cloud computing. In this section, variety of task scheduling technique is reviewed briefly. These all task scheduling techniques are for different intentions.

A. Opportunistic Load Balancing (OLB)

OLB schedule each task, in random order, to the next expected to be available machine, regardless of the task's expected execution time on that machine [8].The intention behind OLB is to keep all machines as busy as possible. One advantage of OLB is its simplicity, but because OLB does not consider expected task execution times, so the mappings it finds can result. Even though OLB keeps all resources as busy as possible, the main problem of OLB is poor makespan due to the expected task execution time of task does not consider by OLB.

B. Minimum Execution Time (MET)

In contrast to OLB, Minimum Execution Time (MET) assigns each task, in random order, to the machine with the minimum expected execution time for that task, regardless of that machine's availability[9].The aim behind MET is to give each task to its best machine. This can cause a severe load imbalance across machines. MET gives each task to its best machine; drawback of this algorithm is severe load imbalance across machines.

C. Minimum Completion Time (MCT)

Minimum Completion Time (MCT) assigns each task, in random order, to the machine that has the minimum expected completion time for that task. Due to this some tasks to be assigned to machines that do not have the minimum execution time for them [10]. The intension of MCT is to combine the benefits of OLB and MET, while avoiding the circumstances in which OLB and MET perform poorly. In MCT, some tasks to be assigned to machines that do not have the minimum execution time for them.

D. Min-min Task Scheduling Algorithm

The Min-min heuristic begins with the set of all unmapped tasks (Set U). Then, the set of minimum completion times, M, for each task $t_i \in U$, is found. More over the task with the overall minimum completion time from M is selected and assigned to the corresponding machine (Thus named Min-min). At Last, the newly mapped task is deleted from U, and the process repeats until all tasks are mapped (i.e., U is empty).[13] However, Min-min considers all unmapped tasks during each mapping decision and MCT only considers one task at a time. Min-min maps the tasks in the order that changes the machine availability status by the least amount that any assignment could. Let t_i be the first task mapped by Min-min onto an empty system. The machine that finishes t_i the earliest, say m_j, is also the machine that executes t_i the fastest. For every task that Min-min maps after t_i, the Min-min heuristic changes the availability status of m_j by the least possible amount for every assignment. Therefore, the percentage of tasks assigned to their first choice (on the basis of execution time) is likely to be higher for Min-min than for Max-min (defined next). The expectation is that a smaller makespan can be obtained if more tasks are assigned to the machines that complete them the earliest and also execute them the fastest. Hear the main problem is regarding as Min-min gives higher priority to small tasks, it increases Response time for large tasks.[11]

E. Max-min Task Scheduling Algorithm

In reference to the heuristic which we have seen previously i.e. .Min-min Algorithm, the Max-min heuristic is also very similar to it. The Max-min heuristic begins with the set of unmapped tasks. Let U be the set of all unmapped tasks. Then, the set of minimum completion times, M, is found. The heuristic then selects overall maximum completion time from M and assigns to the resembling machine. And so it is named as Max-min Algorithm. Finally, the new mapped task is removed from U, and the process keeps on repeating until all tasks are mapped which implies till U is empty.[8] Intuitively, Max-min tries to minimize the penalties incurred from performing tasks with longer execution times. For instance, let the meta-task being mapped has many tasks. Assume that one task has short execution time and one has a very long execution time. Mapping the task with the longer execution time to its best machine first allows this task to be executed concurrently with the remaining tasks (with shorter execution times). For this case, this would be a better mapping technique than a Min-min mapping. Because in Min-min mapping all of the shortest jobs (tasks) would execute first, and then the longer running task would execute while several machines sit idle. Thus, in cases similar to this example, the Max-min heuristic improves makespan. It may give a mapping with a more balanced load across machines. [13]

F. Load Balanced Min-min Algorithm

The traditional Min-Min algorithm is a simple algorithm that produces a schedule that minimizes the makespan than the other traditional algorithms in the literature. But it fails to produce a load balanced schedule. Load Balanced Min-Min (LBMM) algorithm is proposed that reduces the makespan and increases the resource utilization. The proposed method has two-phases. In the first phase the traditional Min-Min algorithm is executed and in the second phase the tasks are rescheduled to use the unutilized resources effectively[12].

The Min-Min algorithm first finds the minimum execution time of all tasks. Then it chooses the task with the least execution time among all the tasks. The algorithm proceeds by assigning the task to the resource that produces the minimum completion time. The same procedure is repeated by Min-Min until all tasks are scheduled. The limitation of Min-Min algorithm is that it chooses smaller tasks first which makes use of resource with high computational power. As a result, the schedule produced by Min-Min is not optimal when number of smaller tasks exceeds the large ones. To overcome this difficulty, Max-min algorithm schedules larger tasks first [7]. But in some cases, the makespan may increase due to the execution of larger tasks first. The waiting time of smaller tasks is also increased in Max-Min. The algorithm proposed in this paper outperforms all those algorithms both in terms of makespan and load balancing. Thus a better load balancing is achieved and the total response time of the

grid system is improved. The proposed algorithm applies the Min-Min strategy in the first phase and then reschedules by considering the maximum execution time that is less than the makespan obtained from the first phase. So LBMM executes Min-Min in the first round. In the second round it chooses the resources with heavy load and reassigns them to the resources with light load.

III . THEORITICAL ANALYSIS

TABLE 1 : COMPARISON OF TASK SCHEDULING ALGORITHM BASED ON DIFFERENT FACTORS

	OLB	**MET**	**MCT**	**Min-Min**	**Max-Min**	**LBMM**
Method	Batch Mode					
Makespan	Very Poor	< OLB But > Rest of all	< OLB and MET	< OLB and MET But Sometimes > MCT	< OLB and MET But Sometimes > MCT	< Rest of all
Time Complexity	O(n) Where n = no. of tasks	O(n)	O(n)	$O(mn^2)$ Where m = no. of resources	$O(mn^2)$	$O(mn^2)$
Speed	√	√	√	□	□	√
Inspiration	Simplicity and Resource Utilization	Select best machine	Use minimum completion criteria	Overall response time is decrease	Less response time for large tasks	Balance Load across Resources

IV. CONCLUSION

Scheduling is the major issue in the management of tasks execution in cloud computing. In this paper, we have surveyed the various existing task scheduling algorithm and tabulated their parameters such as time complexity, makespan, speed and method. In addition to these algorithms, there are many studies related to and based on them that are searching for improvements, optimization, load balance and etc. There are many aspects could be considered as topics of research to introduce more accurate and improved algorithms rather that those introduced here such as cost of the task execution, arriving rate of the tasks on each of the resource, cost of the communication etc.

V. ACKNOWLEDGEMENT

Gaurang Patel would like to thank to his thesis guide Prof. Rutvik Mehta for his great effort and instructive comments in this paper work. Lastly, I wish to thank to all those who helped me during the lifetime of my research work.

VI. REFERENCES

[1] Salim Bitam, "Bees Life algorithms for job scheduling incloud computing", International Conference on computing and Information Technology, 2012.

[2] The NIST Definition of Cloud Computing, Peter Mell Timothy Grance NIST Special Publication 800-145.

[3] Michael Armbrust, Armando Fox, Rean Griffith, Anthony D. Joseph, Randy Katz, Andy Konwinski, Gunho Lee, David Patterson, Ariel Rabkin, Ion Stoica, and Matei Zaharia "Above the Clouds: A Berkeley View of Cloud Computing", EECS Department,University of California, Berkeley ,Technical Report No. UCB/EECS-2009-28, February 10, 2009.

[4] Mythry Vuyyuru, Pulipati Annapurna, K.Ganapathi Babu, A.S.K Ratnam, "An Overview of Cloud Computing Technology", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-3, July 2012

[5] David Burford, CFA, MBA, MCP, "Cloud Computing: A Brief Introduction", LAB Enterprises Inc, February 20, 2010.

[6] L. Wang, G. Laszewski, M. Kunze and J. Tao, "Cloud computing: a perspective study, J New Generation Computing", 2010, pp 1-11

[7] Sun Microsystems, "Introduction to cloud computing architecture". White Paper, Sun Microsystems, June 2009.

[8] AnjuBala, Dr.InderveerChana, "A Survey of Various Workflow Scheduling Algorithms in Cloud Environment", 2nd National Conference on Information and Communication Technology (NCICT) 2011.

[9] O. M. Elzeki, M. Z. Reshad, M. A. Elsoud,"Improved Max-min algorithm in Cloud Computing", International Journal of Computer Applications (0975-8887) Volume 50 – No. 12, July 2012.

[10] T.Casavant and J.Kuhl, "A Taxonomy of Scheduling in General Purpose Distributed Computing Systems", IEEE Trans. On Software Engineering, vol.14, no.3, February 1988, pp.141-154.

[11] Shu-Ching Wang, Kuo-Qin Yan *(Corresponding author), Wen-Pin Liao and Shun-Sheng Wang "Towards a Load Balancing in a Three-level Cloud Computing Network", Institute of Electrical and Electronics Engineers - 2010.

[12] T. Kokilavani, Dr. D.I. George Amalarethinam, "Load Balanced Min-Min Algorithm for Static Meta-Task Scheduling in Grid Computing", International Journal of Computer Applications (0975 – 8887),Journal of Computer Applications (0975 – 8887), Volume 20– No.2, April 2011

