INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X

(Volume2, Issue4)

Available online at: www.ljariit.com

The Art Of Scheduling In Cloud Computing

Harshita Vashishth
Information Technology Department
Maharishi Markandeshwar University
Mullana, Haryana
harshita.vashishth@yahoo.com

Kamal Prakash
Associate Professor
Information Technology Department
Maharishi Markandeshwar University
Mullana, Haryana

Abstract: Cloud computing is one of the fastest growing technologies which has replaced machine paradigm shift. Cloud computing provides very large scalable and virtualised resources over Internet. In Cloud computing, there are many jobs that are required to be executed by available resources while achieving best performance, minimal total time for completion, shortest response time, utilization of resource etc. To achieve these objectives we need to design, develop and propose a scheduling algorithm. In this paper we are surveying various types of scheduling techniques and issues related to them in Cloud computing. Here we have also surveyed various existing algorithms to find their appropriation according to our needs and their shortcomings.

Keywords: Cloud Computing, Scheduling, Algorithm, Quality of Service, Job Scheduling.

I. INTRODUCTION

Cloud computing is one of the fastest growing technologies which has replaced machine paradigm shift [1]. The name "cloud" is based on the pictorial depiction in computer network diagrams and is an abstraction that conceals complex infrastructure. The main advantage of Cloud computing is On-Demand gathering of information, services and products. Cloud computing provides end users as "a pay as you go model". That is, we can use as much or as less we want to use, use when we want to and pay for same accordingly. The customer demands for available services according to their desired need for Quality of Service. Cloud Providers provides services to a customer and charges according to the usage. Cloud computing is a technique that allows us to use resources that actually reside at a different location than ours. The increasing growth in virtualization and Cloud computing technologies shows the number of jobs is increasing nowadays, that require services of this virtual machine. Job Scheduling is one of the most important activities performed in Cloud computing environment. Job Scheduling efficiently increases the working of Cloud computing environment which ultimately leads to maximum profit. Job Scheduling is used to allocate a particular job to particular resources in a particular time. The goal of scheduling algorithms is distributing the load on processors and maximizing their utilization while minimizing the total job execution time. Other goals of job-scheduling algorithms are better quality of service, maintaining efficiency, high throughput etc. there are two categories of scheduling algorithms: Static Scheduling Algorithm and Dynamic Scheduling Algorithm. Both have their own advantages and disadvantages.

The remainder of the paper is organized as follows. Section II presents the Types of Scheduling. Section III describes Existing Scheduling Algorithms; Section IV describes comparison between parametric evaluations between Existing Scheduling Algorithms. Finally, Section V specifies the conclusion.

II. SCHEDULING

In Cloud computing, job scheduling problem is the biggest and crucial issue. Traditional job scheduling algorithms are not able to provide scheduling in Cloud computing environment. An efficient job scheduling strategy aims at yielding less response time such

that execution of submitted jobs completes within minimum possible time and resources can be reallocated. There are different types of scheduling based on different criteria, such as:

a) Static scheduling

In Static scheduling, all the information about jobs and available resources is known in advance, after which the job is assigned to resources [2].

b) Dynamic scheduling

In dynamic scheduling, jobs are allocated to resources at runtime by the scheduler. It is more flexible than static scheduling, helps in determining run time in advance. The overhead is more as compared to static scheduling [2].

c) Online mode scheduling

In online mode scheduling, the resources are allocated to a job the moment it arrives, on the basis of available resources at that very moment by the scheduler [3].

d) Batch mode scheduling

In Batch mode scheduling, the scheduler allocates resources to jobs in batches that is, it piles up jobs over a period of time and then their execution starts after a specific time interval. It is also called offline scheduling [3].

e) Pre- Emptive scheduling

In Pre- Emptive scheduling, a job can be interrupted during its execution and a job can be migrated to another resource other than its originally allocated resources, available at the moment. The consideration of constraints such as priority makes it more helpful. [4][5][6].

f) Non Pre- Emptive scheduling

In Non Pre- Emptive scheduling, resources are not allowed to be re-allocated until the currently running jobs are finished. [4][5][6].

For Job scheduling various algorithms are used. Some of the Traditional algorithms used are:

a) FIFO Algorithm

In this algorithm, first arrived job in the queue is processed first. It is simple and fast algorithm [1][3].

b) Round Robin algorithm

In this algorithm, processes are dispatched in FIFO manner and a limited amount of CPU time called time slice is provided. If the process is not completed in the assigned time then the CPU is allocated to job in the waiting queue and the cycle goes on. [5][6].

c) Min-Min Algorithm

In this algorithm, small jobs are executed first, which in turn delays the longer jobs for longer period of time.

d) Max-Max Algorithm

In this algorithm, larger jobs are executed first which in turn delays small jobs for longer period of time.

e) Most fit task scheduling Algorithm

In this algorithm, job which is best fit in the queue is executed first.

III. EXISTING SCHEDULING ALGORITHMS

A. Optimistic Job Scheduling Algorithm

Shalmali, et al [7] has proposed an Optimistic Job Scheduling Algorithm for Cloud Computing environments. In this paper, an application is designed for scheduling of requests. Non Pre-Emptive queuing model is used to handle the requests. The web application created performs two operations: Uploading a file and Downloading a file. For this three folders are used by them. When uploading or downloading is performed, the size of the folder is analyzed for load balancing. It achieves users QoS requirements and service provider's maximum profit.

B. Priority based Job scheduling Algorithm:

Shamsollah, et al [8], proposed a new scheduling algorithm on multi-criteria and multi-decision priority driven scheduling algorithm. This scheduling algorithm consists of three level of scheduling: object level, attribute level, alternate level. The object level is

scheduling level, the attribute level is resource level and Job level is alternate level. Analysis of proposed algorithm discusses 3 issues: complexity, consistency and makespan. In this algorithm, priority can be set by job resource ratio. Then priority vector can be correlated with each queue. This algorithm has high throughput and less finish time.

C. Cost-Based Task Scheduling Algorithm

Selvarani, et al [9], proposed an improved cost-based scheduling algorithm for efficient mapping of tasks to available resources in cloud. In this paper, a task is scheduled based on cost of resources and computation performance. In this algorithm, tasks are sorted on the basis of priority and stored in low, medium and high priority levels. It improves computation/communication ratio.

D. Reliable scheduling algorithm

Arash, et al [10] proposed a reliable scheduling algorithm in cloud computing environment. This RSDC (Reliable Scheduling Distributed in Cloud Computing) Algorithm is based on PPDD (Processor-set Partitioning and Data Distribution) Algorithm. The PPDD Algorithm discusses about general load balancing and scheduling with multiple loads. In RSDC Algorithm, major job is divided into sub-jobs. To balance the jobs, the request and acknowledgement time are calculated separately. Scheduling of each job is done by calculating request and acknowledgement time in the form of shared job. So that efficiency of the system is increased.

E. Max-Min Scheduling Algorithm using Petri Net

El-Sayed, et al [11], proposed a new algorithm based on impact of RASA Algorithm. The improved Max-Min Algorithm is based on predictable execution time instead of complete time as a selection basis. Petri nets are used to represent the concurrent behaviour of distributed systems. Max-Min shows achieving schedule with comparable lower makespan rather than RASA and original Max-Min.

F. Priority based Service Scheduling

Dakshayini, et al [12], proposed new scheduling algorithm based on priority and admission control. In this algorithm, each admitted queue is assigned a priority. Admission of each queue is decided by calculating tolerable delay and service cost. Advantage of this algorithm is that this policy with the future cloud architecture has achieved very high (99%) service achievement rate with definite QoS. As this policy provides the highest precedence for highly paid user service-requests, overall servicing cost for the cloud environment increases.

G. Resource Aware Scheduling

Parsa, et al [13], proposed a new task scheduling algorithm RASA. It is formed by two traditional scheduling algorithms: Max-Min and Min-Min. RASA uses advantages of both Max-Min and Min-Min algorithms and does not include their disadvantages. The experiment result shows RASA is outstanding in performance as compared to existing scheduling algorithms for large scale distributed systems.

TABLE COMPARISION: Difference between Various Existing Algorithms According To Standard Parameters.

Scheduling Algorithm	**Scheduling Method**	**Scheduling Parameters**	**Scheduling Factors**	**Findings**	**Environment**
Optimistic Differentiated Job Scheduling System For Cloud Computing	Dependency Mode	Quality of Service, Maximum Profit	Single Job with Multiple User	The QoS needs of the Cloud Computing user and the Maximum Profits of the Cloud Computing Service Providers are Accomplished.	Cloud Environment
Improved Cost-Based Algorithm For Task Scheduling In Cloud Computing	Batch Mode	Cost, Performance	Unscheduled Task Group	1. Measures both Resource Cost and Computation Performance.	Cloud Environment

				2. Improves the Computation/Communication Ratio.	
A Priority Based Job Scheduling Algorithm In Cloud Computing	Dependency Mode	Priority to each Queue	An Array of Job Queue	Less Finish Time	Cloud Environment
RSDC (RELIABLE Scheduling In Cloud Computing)	Batch Mode	Processing Time	Grouped Task	1. It is used to Reduce Processing Time. 2. It is Efficient for Load Balancing.	Cloud Environment
An Optimal Model For Priority Based Service Scheduling Policy For Cloud Computing Environment	Batch Mode	Quality of Service, Service Request Time	An Array of Workflow Instances	1.High QoS 2.High Throughput	Cloud Environment
Extended Max-Min Scheduling Using Petri Net And Load Balancing	Batch Mode	Load Balancing, Finishing Time	Grouped Task	1. It is used for Efficient Load Balancing. 2. Petri Net is used to remove Limitation of Max-Min Algorithm.	Cloud Environment
RASA (Resource Aware Scheduling Algorithm)	Batch Mode	Makespan	Grouped Task	It is used to Decrease Makespan	Grid Environment

V CONCLUSION

Scheduling is one of the most important issues in cloud computing environments. In this paper, we have analysed various existing algorithms in cloud computing and tabulated various parameters. The Existing Algorithms have high throughput and cost effective but they do not consider factors such as reliability and availability. So there is a need to implement an algorithm that improves availability and reliability in cloud computing environment.

REFERENCES

[1]. Gan Huo-Ning, Huang Ting-Lei, Gao Shuai, "Genetic Simulated Annealing Algorithm For Task Scheduling Based On Cloud Computing Environment", IEEE, 2010, pp.60-63.
[2]. Thomas A. Henzinger, Anmol V. Singh, Vasu Singh, Thomas Wies, "Static Scheduling In Clouds"
[3]. Yun-Han Lee Et Al, Improving Job Scheduling Algorithms in A Grid Environment, Future Generation Computer Systems, 27 (2011) pp.991-998
[4]. Fatos Xhafa, Ajith Abraham, "Computational Models and Heuristic Methods for Grid Scheduling Problems", Future Generation Computer Systems, 26 (2010) pp. 608-621.
[5]. Types of Scheduling: http://orzhovgilden.site11.com/projects/CPU.html.
[6]. Types of Scheduling Algorithms: http://webdocs.cs.ualberta.ca/~tony/C379/Notes/PDF/05.4.pdf
[7]. Shalmali Ambike, Dipti Bhansali, Jaee Kshirsagar, Juhi Bansiwal "An Optimistic Differentiated Job Scheduling System For Cloud Computing" Internal Journal Of Engineering Research And Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol.2,Issue 2, Mar-Apr 2012, pp.1212-1214.

[8]. Shamsollah Ghanbari, Mohamed Othman "A Priority Based Job Scheduling Algorithm In Cloud Computing" International Conference On Advances Science And Contemporary Engineering, 2012 (ICASCE 2012, pp.147-152).

[9]. Mrs. S. Selvarani, Dr. G. Sudha Sadhasivam, "Improved Cost-Based Algorithm for Task Scheduling in Cloud Computing", IEEE 2010, pp.1-5.

[10]. Arash Ghorbannia Delavar, Mahdi Javanmard, Mehrdad Barzegar Shabestari and Marjan Khosravi Talebi "RSDC (Reliable Scheduling Distributed In Cloud Computing)" International Journal of Computer Science, Engineering and Applications (IJCSEA) Vol.2, No.3, June 2012, pp: 1-16.

[11]. El-Sayed T. El-Kenawy, Alibraheem El- Desoky, Mohamed F. Al-Rahamawy "Extended Max-Min Scheduling Using Petri Net And Load Balancing" International Journal Of Soft Computing And Engineering (IJSCE) ISSN:2231-2307, Vol.2,Issue-4,Sept.2012.

[12]. Dr. M. Dakshayini , Dr. H. S. Guruprasad "An Optimal Model For Priority Based Service Scheduling Policy For Cloud Computing Environment" International Journal Of Computer Applications (0975-8887)Vol.32- No.9.,Oct 2011,pp.23-29.

[13]. Saeed Parsa and Reza Entezari-Maleki, "RASA: A New Task Scheduling Algorithm in Grid Environment" In World Applied Sciences Journal 7 (Special Issue of Computer & IT):152-160, 2009

