A Survey on Different Types of Cloud-Based Scheduling Algorithms

Jalpa patel[1]
[1]M.E, Computer Engineering,
[1]Alpha College of Engineering, Gujarat, India

Abstract—*Cloud computing is a type of computing that relies on sharing computing resources where the customer need not own the necessary infrastructure and pay for only what they use. Cloud Computing refers to the use of computing, platform, software, as a service. Computing resources are delivered as virtual machines. In such a scenario, job scheduling algorithms play an important role where the aim is to schedule the job effectively so as to reduce the execution time and improve resource utilization. This paper presents an attempt is made to address the various types of scheduling in the cloud computing environment.*

Keywords : cloud computing, scheduling, algorithm.

I. INTRODUCTION

According to Buyyaet. al. [1] a cloud is a type of parallel and distributed system consisting of a collection of interconnected and virtualized computers that are dynamically provisioned and presented as one or more unified computing resources based on service-level agreement. Cloud computing is the use of computing resources that are delivered as a service over the network. Cloud computing means a type of Internet-based computing, where different services (such as servers, storage and applications) are delivered to an organization's computers and devices through the Internet computing, the word "cloud" is used for "the Internet". The two most significant components of cloud computing architecture are known as the front end and the back end. The front end is the part seen by the client, i.e. the computer user. This includes the client's network (or computer) and the applications used to access the cloud via a user interface such as a web browser. The back end of the cloud computing architecture is the 'cloud' itself, comprising various computers, servers and data storage devices. The need for scheduling is to allocate the appropriate tasks at the right time in the desired environment. Job scheduling system is one of the core and challenging issues in a Cloud Computing system. The aim of Job scheduling systems in Cloud or Grid computing mainly considers how to meet the QoS requirements.

Scheduling process in cloud can be generalized into three stages namely–

- Resource discovering and filtering – Datacenter Broker discovers the resources present in the network system and collects status information related to them.
- Resource selection – Target resource is selected based on certain parameters of task and resource. This is deciding stage.
- Task submission -Task is submitted to resource selected.

The simplified scheduling steps mentioned above are shown in Figure 1

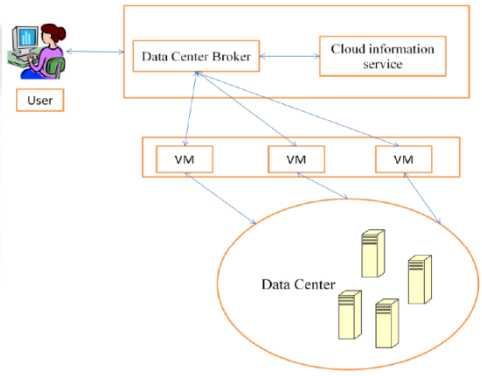

Fig. 1 Scheduling Process

The rest of this paper is organized as follows: Section 2 briefly discusses the different types of scheduling in cloud computing. Section 3 present existing scheduling algorithms. Section 4 present comparisons of scheduling algorithms. In Section 5 concludes the paper.

II. TYPES OF SCHEDULING

A. Static vs. Dynamic scheduling

1) Static scheduling

There is a pre scheduling jobs in Static scheduling. Information regarding all resources as well as all the tasks in an application is assumed to be available by the time the application is scheduled. No job failure and resources are assumed available all the time.

2) Dynamic scheduling

In Dynamic scheduling Jobs are dynamically available for scheduling over time by the scheduler with no issues, to be able of determining run time in advance. The dynamics of job execution, which refers to the situation when job execution could fail due to some resource failure or stopped due to the arrival in the system of high priority jobs when the case of preemptive mode is considered. The dynamics of resources in which workload on resources can significantly vary over time.

B. Centralized, Decentralize vs. Hierarchical scheduling

These types of scheduling differ in the control of the resources and knowledge of the overall system.

1) Centralized scheduling

In Centralize scheduling more control on resources. The scheduler has knowledge of the system by monitoring of the resource state.

Advantages: ease of implementation, efficiency and more control and monitoring on resources.

Disadvantages: lack scalability, fault tolerance and efficient performance

2) Decentralize scheduling

There is no central entity controlling the resources. Scheduling decisions are shared by multiple distributed

schedulers. Less efficient then centralized scheduling.

3) Hierarchical scheduling

In hierarchical scheduling one entity coordinate different scheduler at a certain level. Schedulers at the lowest level in the hierarchy have knowledge of the resources.
Disadvantages: lack of scalability and fault tolerance

C. Preemptive vs. Non preemptive scheduling

1) Preemptive scheduling

This scheduling criterion allows each job to be interrupted during execution and a job can be migrated to another resource leaving its originally allocated resource unused to be available for other jobs. It is more helpful if there are constraints as priority to be considered.

2) Non preemptive scheduling:

In which resource are not being allowed to be re allocation until the running and scheduled job finished its execution.

D. Immediate/online vs. Batch / offline mode

1) Immediate/online mode

In which scheduler schedules any recently arriving job as soon as it arrives with no waiting for next time interval on available resources at that moment.

2) Batch / offline mode

The schedule holds arriving jobs as group of problems to be solved over successive time intervals. So that it is better to map a job for suitable resources depending on its characteristics.

E. Independent vs. Workflow scheduling

1) Independent scheduling

Tasks are independent to each other. Tasks are run independently to each other.

2) Workflow scheduling

Tasks are dependent on each other. Dependency means there are precedence orders existing in tasks, that is , a task can not start until all its parent are done. Workflows are represented by Directed Acyclic Graph notation. Each task can start its execution only when all preceding tasks in DAG are already finished.

III. EXISTING SCHEDULING ALGORITHM

The Following scheduling algorithms are currently prevalent in clouds.

A. Improving scheduling of backfill algorithms using balanced spiral method for cloud metascheduler [2].

Suresh.A and Vijayakarthick.P propose an improved backfillalgorithm (IBA) using balanced spiral method. This paper addresses the problem of making dynamic metascheduler in cloud environment. The multiple objectives are maximizing the resource utilization and minimize the resource gap of idle resources.

B. Improve Cost-base Algorithm for Task Scheduling in Cloud computing [3].

Mrs.S.Selvarani, Dr.G.SudhaSadhasivam propose improve cost base algorithm. The objective of this paper is to schedule task groups in cloud computing platform, where resources have different resource costs and computation performance. Due to job grouping, communication of coarse-grained jobs and resources optimizes computation / communication ratio. For this purpose, an algorithm based on both costs with user task grouping is proposed.

C. A Dynamic Job Grouping-Based Scheduling for Deploying Applications with Fine-Grained Tasks on Global Grids [4].

NithiapidaryMuthuvelu, Junyang Liu, Nay Lin Soe, SrikumarVenugopal Anthony Sulistio and RajkumarBuyya propose DJGSDA. The overall processing undertaking of these applications involves high overhead time and cost in terms of (i) job transmission to and from Grid resources and, (ii) job processing at the Grid resources. Authors present a scheduling strategy that performs dynamic job grouping activity at runtime and convey the detailed analysis by running simulations. The job grouping is done based on a particular granularity size. Granularity size is the time within which a job is processed at the resources

D. A Secure Resource and Job scheduling Model with Job Grouping strategy in Grid Computing [5].

Raksha Sharma, Vishnu Kant Soni, ManojKumarMishra, Sarita Das propose SRJ. SRJ performs job grouping activity at runtime and the simulation results shows significant improvement in the processing time of jobs and Resource utilization. The model is divided into three levels: user level, global level and cluster level. In user level, the grid users submit their jobs into grid environment. Global level scheduler receives jobs and makes a queue and sends these jobs for execution according to the capability of available resources at the cluster level, in cluster level, jobs are computed and results are sent back to the users. The global and local scheduler interacts with each other to make an optimal scheduling of jobs.

E. Priority-Based Consolidation of Parallel Workloads in the Cloud [6].

Xiaocheng Liu, Chen Wang, Bing Bing Zhou, JunliangChen,Ting Yang, and Albert Y. Zomaya, focus on improving resource utilization for data centers that run parallel jobs, particularly they intend to make use of the remaining computing capacity of data center nodes that run parallel processes with low resource utilization to improve the performance of parallel job scheduling.

Basic Algorithms used
CMBF (Conservative Migration supported BackFilling) requires tracking backfilling jobs for each job in the queue when making preemption decisions.
AMBF (Aggressive Migration Supported BackFilling) only tracks backfilling jobs for the job at the head of the queue and allows the head-of queue job to preempt other jobs.
Author leverage virtualization technologies to partition the computing capacity of each node into two tiers, the foreground virtual machine (VM) tier (with high CPU priority) and the background VM tier (with low CPU priority). They provide scheduling algorithms for parallel jobs to make efficient use of the two tier VMs to improve the responsiveness of these jobs.

F. Max-Min Task Scheduling Algorithm [7].

Task scheduling algorithm [7] is responsible for mapping jobs submitted to cloud environment onto available resources in such a way that the total response time, the

makespan is minimized. Task scheduling process is an allocation of one or more time intervals to one or more resources. In cloud computing, the problem of scheduling a set of submitted tasks from different users on a set of computing resources to minimize the completion time of a specific task or the makespan of a system.

Step 1: The Max-min algorithm is commonly used in distributed environment which begins with a set of unscheduled tasks.

Step 2: Then calculate the expected execution matrix and expected completion time of each task on the available resources.

Step 3: Choose the task with overall maximum expected completion time and assign it to the resource with minimum overall execution time.

Step 4: Finally recently scheduled task is removed from the meta-tasks set, update all calculated times, then repeat until meta-tasks set become empty.

The algorithm focuses on minimizing the total makespan which is the total complete time in large distributed environment.

IV. COMPARISION

Scheduling Algorithm	Scheduling Type	Scheduling Parameter	Achievements	Environment
Improving scheduling of backfill algorithms using balanced spiral method for cloud metascheduler [2]	Centralize, Dynamic and non-preemptive Scheduling	Utilization, Processing time	Maximize resource utilization, Reduce processing time	Cloud Environment
Improve Cost-base Algorithm for Task Scheduling in Cloud computing [3].	Independent and Dynamic Scheduling	Execution time, Processing cost	Reduce Execution time, Processing cost	Cloud Environment
A Dynamic Job Grouping-Based Scheduling for Deploying Applications with Fine-Grained Tasks on Global Grids [4].	Immediate and Dynamic scheduling	Processing time, Processing cost	Reduce Processing time, Processing cost	Grid Environment
A Secure Resource and Job scheduling Model with Job Grouping strategy in Grid Computing [5].	Decentralize, Dynamic Scheduling	Execution time, Processing cost	Reduce execution time, Processing cost	Grid Environment
Priority-Based Consolidation of Parallel Workloads in the Cloud [6].	Preemptive, Dynamic scheduling	Response time, Migration, utilization	Increase response time , reduce migration and increase utilization	Cloud Environment
Max-Min Task Scheduling Algorithm [7].	Static scheduling	Makespan	Minimize the total makespan	Cloud Environment

Table 1:A Comparative Study On Cloud Based Scheduling Algorithms

V. CONCLUSION

This study has presented that there are various types of scheduling algorithms which work on various parameters. There exist several applications with a large number of lightweight jobs. The overall processing undertaking of these applications involves high overhead time and cost in terms of job transmission to and from cloud resources and job processing at the cloud resources. Existing scheduling algorithm gives high throughput and cost effective but they do not consider high overhead time. So we need algorithm that overcome the problem of high overhead time and cost in terms of job transmission to and from Cloud resources.

REFERENCE

[1] Buyya, Rajkumar, et al. "Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility."Future Generation computer systems 25.6 (2009): 599-616.

[2] Suresh, A., and P. Vijayakarthick. "Improving scheduling of backfill algorithms using balanced spiral

method for cloud metascheduler."Recent Trends in Information Technology (ICRTIT), 2011 International Conference on.IEEE, 2011.

[3] Selvarani, S., and G. SudhaSadhasivam. "Improved cost-based algorithm for task scheduling in cloud computing." Computational Intelligence and Computing Research (ICCIC), 2010 IEEE International Conference on.IEEE, 2010.

[4] Muthuvelu, Nithiapidary, et al. "A dynamic job grouping-based scheduling for deploying applications with fine-grained tasks on global grids." ACSW Frontiers 5 (2005).

[5] Sharma, Raksha, et al. "A Secure Resource and Job scheduling model with Job Grouping strategy in Grid computing." Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on.Vol. 4. IEEE, 2010..

[6] Liu, Xiaocheng, et al. "Priority-Based Consolidation of Parallel Workloads in the Cloud." (2012): 1-1.

[7] Elzeki, O. M., M. Z. Reshad, and M. A. Elsoud. "Improved Max-Min Algorithm in Cloud Computing." International Journal of Computer Applications 50.12 (2012): 22-27.