

Job Scheduling Algorithms in Cloud Computing: A Survey

Himanshu Goel
MTech Research Scholar
CSED, GBPEC Ghurdauri

Narendra Chamoli
MTech Research Scholar
CSED, GBPEC Ghurdauri

ABSTRACT

Cloud Computing is the new IT Paradigm that makes the delivery of computing resources (Hardware and Software), applications and data as a service over the internet to its users. Cloud computing mainly intended to provide reliable, dynamic and virtualized services in terms of resources for doing computation, storage and knowledge sharing. An essential requirement in cloud computing is scheduling of current jobs to be executed within some given metrics or constraints. In Cloud computing, execution of jobs requires various resources which are available to them by fulfilling certain constraints like best performance, minimum execution time, shortest response time, fault-tolerance and quality of expected services. The scheduler should order the jobs in a way where the balance between improving the quality of services and at the same time maintaining the efficiency and fairness among the jobs. Thus, in large-scale distributed systems, the performance evaluation of the algorithm is important. In this paper, our major goal is to study systematic review of various job scheduling algorithms.

Keywords

Cloud Computing, Job Scheduling, Scheduling Algorithms

1. INTRODUCTION

In general, a cloud is a collection of clusters (parallel and distributed system consisted of bunch of interconnected and virtualized computers) that are dynamically changing and a whole of them view as a unified computing resources mutually agreed upon a Service- level agreement. Cloud Computing is the use of computing resources (Hardware and Software) that are delivered as a service over a network to the users.

The US National Institute of Standards and Technology (NIST) define the cloud computing in most common terms covering all its aspects. The NIST defines cloud computing as [1]:

“Cloud computing is a model 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.”

Cloud computing is a new IT paradigm in which dynamic resource allocation is taking place on consolidated resources from many techniques for parallel computing, distributed computing and platform virtualization technologies. Cloud computing is a type Distributed Computing paradigm where a large pool of systems are connected in private or public networks providing infrastructure for application, data and

file storage which can be dynamically scalable. Cloud providers provide services to their customers. The resources used for these services can be metered and the users can be charged only for the resources they used. Cloud computing is an environment in which we use the computing resources in distant data center rather than the local computing systems. The cloud environment provides different virtualized platforms that help user to accomplish their jobs with minimum completion time and minimum costs.

Job Scheduling is a critical component of the cloud resource management. Job Scheduling is a process of how jobs should be executed in the environment in terms of actual mapping of resources and time when the jobs get executed. It is mainly responsible for resource sharing or multiplexing at several levels. For example, a server can be shared among many virtual machines, each virtual machine may support many applications and each application may consist of many threads.

Scheduling process in the cloud can be done in three stages:

- Resource discovering and filtering – Datacenter Broker know the current status of all the resources that are available in the cloud and also the remaining resources that may be available. Actually, these resources are generally the Virtual machines. It regularly collects the status of each resource attached to the cloud.
- Resource selection - Based on information obtained from the resources status regarding current queued jobs and information on the status of cloud resources, the cloud scheduler makes decisions regarding the creation or deletion of specific cloud nodes (VMs) in order to best suit the set of jobs waiting to be run.
- Job Submission – In this stage, finally the job is submitted to best available resource selected.

The objective of this paper is to be focus on various scheduling algorithms. The rest of the paper is organized as follows. Section 2 presents work on scheduling in cloud environment. Section 3 presents the paper with some conclusion.

2. SCHEDULING

Job scheduling problem in Cloud is as “Jobs and Resources need to be allocated and scheduled in such a way that Cloud Users can complete their jobs with minimal time (within the deadline) and cost and maximize the user satisfaction and throughput of Cloud Resource Provider [2]”. In a broader sense, the user is expected to complete his/her jobs with minimum time and minimum cost.

A. Major Performance Metrics of scheduling Algorithm:-

- **Makespan** – It is the total length of the Schedule means when all the jobs on schedule get finished.
- **Execution Cost** – It is defined as the total cost of all the resources used at the execution of the job.
- **Job Rejection Ratio** – It is defined as the total rejected jobs due to exceeding of execution time than the deadline time or higher cost to the total number of jobs submitted.
- **Execution Time** – It is defined as the time from when a job is submitted to cloud environment till it gets its final execution.
- **User Satisfaction Level** - It is defined as the how far the services of the service provider give satisfaction in terms of resources like storage and computation.

B. Types of Scheduling:-

1. Static Scheduling v/s Dynamic Scheduling-

In Static Scheduling [3], all the information about the status of all available resources in the cloud as well as all the needs of jobs knowing in advance and after that the job is mapped to suitable resource. No job failure and resources are assumed available all the time.

In Dynamic Scheduling [3], the task allocation is done on the go as the application executes, where it is not possible to find the execution time. The jobs are entering dynamically and the scheduler has to work hard in decision making to allocate resources. The advantage of the dynamic over the static scheduling is that the system need not possess the runtime behavior of the application before it runs.

2. Centralized, Decentralized and Hierarchical Scheduling-

In Centralized Scheduling [4], there is a centralized scheduler or a bunch of many distributed scheduler that have the responsibility to make the global scheduling decisions. In this, there is more control on resources: the scheduler continuously monitors all available resources states and therefore it is easier to obtain efficient schedulers. The advantage is Ease on implementation, but disadvantage is lack of scalability, fault-tolerance and performance.

In Decentralized Scheduling [5], there is no central entity controlling the resources. In this type, the lower schedulers known as local resources machine (LRM) manage and maintain the job queue. It is less efficient than centralized scheduling.

In Hierarchical Scheduling [6], the computation is at three levels. The top level is meta-level, the tasks are not scheduled directly, but reconfiguring the scheduler according to the characteristics of the input jobs. The mediate level is called the group level, where the manager in each group collaborates with each other and allocates jobs for the workers in the group. The bottom level is called the within-group level, where the workers in each group perform self-scheduling.

3. Preemptive vs. Non preemptive scheduling –

In Preemptive Scheduling [7], preemption is allowed; that is, the current execution of the job can be interrupted and the job is migrated to another resource.

In Non preemptive Scheduling [7], a job should entirely be completed in the resource (the resource cannot be taken away from the task, job or application).

4. Immediate vs. Batch Scheduling –

In Immediate Scheduling [8], as soon as the job is arriving, it is scheduled as there will be no waiting for the next time interval.

In Batch Scheduling [8], Jobs are first grouped into some batches, and then they are allocated to the resources by the scheduler.

C. Review of Some proposed Algorithms:

a) BAR: An Efficient Data Locality Driven Task Scheduling Algorithm for Cloud Computing

Jiahui Jin et.al [9] proposed a heuristic Task Scheduling Algorithm known as Balance-Reduce(BAR) in which first the initial task allocation is proposed ,after then the job completion time is gradually reduced by initial task allocation. By collecting the entire global view of all resources state, the algorithm dynamically allocate the data locality. In this system, First author take a set of independent tasks on a homogeneous platform with m tasks and n servers, where each task processes an input block on a server. A job is not completed until all tasks are finished. Cloud computer Clusters workload are mainly design to have an allocation strategy that minimizes the make span of tasks. BAR called Balance-Reduce is a data locality driven task scheduling algorithm, having best case complexity of $O(\max\{m+n, n \log n\} \cdot m)$. BAR is split into two phases, balance and reduce: In Balance phase, a balanced total allocation is produced where all tasks are allocated to their preferred servers evenly. In Reduce phase, we generate a sequence of total allocations, and reduce the make span iteratively. In a poor network environment, BAR tries its best to enhance data locality. When cluster is overloaded BAR decreases the data locality to make tasks start early. The simulation results show that BAR exhibit an improvement and can deal with a large problem instance in a few seconds.

b) Adaptive Resource Allocation for Pre-emptable Jobs in Cloud Systems

Jiayin Li et.al [10] proposed an algorithm based on adaptive resource allocation for the cloud system in which tasks can be pre-emptable. This algorithm adapts the updated states of the actual task executions and adjusts its resource allocation schema accordingly. In this paper, consideration of an infrastructure-as-a-service (IaaS) cloud system is take place. The cloud service providers actually have datacenter which provides computational resources in form of Virtual machines (VM). In Cloud Computing, three different modes of renting the computing resources are available, such as, Advance Reservation (AR), Best-effort and Immediate from a cloud provider. As we have problems in resource utilization, AR and best-effort can be combined. In this paper some of the submitted jobs are in AR mode while other are in Best-effort. Jobs of AR mode have the higher priority than best-effort. So they can preempt the jobs of best-effort. In this paper, two algorithms for the task scheduling: adaptive list scheduling (ALS) and adaptive min-min scheduling (AMMS) [10] are proposed. Once a job is submitted to a scheduler, it will be divided into tasks in the form of Directed Acyclic Graph (DAG). Both ALS and AMMS include a static task scheduling for resource allocation. Then the scheduler will repeatedly re-evaluate the remaining static allocation with a

pre-defined frequency, based on the latest information of task execution. To generate the static allocation two greedy algorithms, the Cloud List Scheduling (CLS) and the Cloud Min-Min Scheduling (CMMS) are used. The final execution time of a task may be different as we expect since there may be race condition for resources or due to high network traffic. The experimental results show that these algorithms work significantly in intense resource contention situation.

c) Improved Cost-Based Algorithm for Task scheduling in Cloud computing

S.Selvarani et.al [11] proposed a technique based on the user task schedule, cost of each task differs. Cloud does not schedule the user task as it is done in traditional ways. In this paper, an improved cost-based scheduling algorithm [11] is proposed in which there actual resources are mapping for jobs are done. This algorithm mainly calculate the cost of resources and computation performance and works on gradually improves the computation/communication ratio by grouping the user tasks by taking consideration of processing capability of a resources and sends the grouped jobs to the resource. In this paper, main focus is on devising a scheduling strategy in which the independent jobs are grouped having small processing requirements and according to network conditions, Scheduler schedules them. In traditional ways, we take user tasks as overhead application but problem is that in doing so, there is no relationship between overhead application and way any tasks create the overhead costs of the resources in cloud system. If complexity of tasks is less but size is large, then cost is high else complexity of task is high and size is short, then cost is low. Activity-based costing measures both the cost of the resources and the computation performance. The cost of every individual resource is different. Tasks are sorted according to their priority, and they are placed in three different lists based on three levels of priority namely high priority, medium priority and low priority. For computation of tasks, the system can take from high priority list first, then medium and then low. The Improved Activity Based Costing method selects a set of resources to be used for computing. It groups tasks according to the processing capability of resources available. The coarse-grained tasks are processed in the selected resources, so that the Computation- Communication ratio is reduced. The experimental results using a simulator show that the time taken to complete tasks after grouping the tasks is very less when compared with time taken to complete the tasks without grouping the tasks.

d) Priority-Based Consolidation of Parallel Workloads in the Cloud

Xiao cheng Liu et.al [12] proposed an algorithm on improving resource utilization for data centers on which jobs execute parallel, particularly when motive is to make use of the remaining computing capacity of data center nodes that run parallel processes with low resource utilization which affects the performance of parallel job scheduling and gradually improves it. Basic Algorithms used CMBF (Conservative Migration supported BackFilling) continuously searches for backfilling jobs for each job in the queue when making preemption decisions. AMBF (Aggressive Migration Supported BackFilling) tracks for all those backfilling jobs for the job that are at the head of the job queue. It will allow the head-of-queue job to preempt the other jobs. Author make use of virtualization technologies to divide 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). By using them,

there is efficient use of two tier VMs to improve the responsiveness for parallel jobs scheduling algorithm.

e) A new Class of Priority-based Weighted Fair Scheduling Algorithm

Li Yang et.al [13] proposed one kind of weighted fair scheduling algorithm. It uses the strict rob priority class which uses an absolute priority queue that are based on the class weighted fair scheduling algorithm (CBWFQ). This algorithm removes the disadvantage of traditional weighted fair scheduling algorithm. In traditional Weighted Fair Scheduling algorithm, based on the weight of each business flow, the services of all active queue is measured and treat accordingly. The job of classifier is to classify the job when a new job arrives. Then buffer is checked for each category and if buffer is not overloaded then job is stored in the buffer otherwise job is dropped. Each job enters a different virtual queue. The four main pillars of this algorithm are Weight, Dispatch, Discard and Rob. Introduction of rob rule together with dropping rule makes it more effective. This new algorithm combined buffer management and queue scheduling. When author works on the real time applications, only then it may cause some delays. It also gave guaranty of fairness and better utilization of buffers. Main advantages are bandwidth allocation and delay without throughput reducibility.

f) A Priority based Job Scheduling Algorithm in Cloud Computing

Shamsollah Ghanbari et.al [14] proposed a new job scheduling algorithm in cloud computing by using mathematical statistics. This algorithm made its foundation on the priority property that why it is known as Priority-Based Algorithm. It is based on multiple criteria decision making model. In 1980 Thomas Saaty was first on to develop a model that make pair wise comparison based on multiple criteria and multiple attributes and named it as Analytical Hierarchy Process (AHP)[15][16]. AHP is purely based on Consistent Comparison Matrix, so by making the use of AHP, comparison matrices are computed according to the attributes and criteria's accessibilities. In this algorithm, each job requests a resource which has a pre-determined priority. So according to resources accessibilities, comparison matrices of each jobs is computed. Author also computes the comparison matrix of resources which will help later for jobs picking. Then author compute priority vectors (vector of weights) for each the comparison matrix and finally a normal matrix of all jobs is computed named as Δ . Similarly, normal matrix of all resources is computed and marks this matrix as γ . The next step of the algorithm is to compute Priority Vector of S (PVS), where S is set of jobs. PVS is calculated by multiplying matrix Δ with matrix γ . At final step, algorithm chooses the job with maximum calculated priority on basis of that suitable resource is allocated to that job. Now the list of jobs is updated and the scheduling process continues till all the jobs are assigned to suitable resource. Experimental results indicate that the algorithm has reasonable complexity. But there some issues such as complexity, consistency and finish time.

g) Agent based Priority Heuristic for Job Scheduling on Computational Grids

Shah et. al [17] proposed job scheduling technique based on agent used for effective and efficient execution of user jobs. This algorithm mainly account on Quality of Service (QoS) parameters like waiting time, turnaround time, response time, total completion time, etc. based on the classifications,

priorities are assigned to each of the job. Agent based Heuristic Scheduling (AHS) uses task agent that makes job distribution more effective to achieve optimum solution. First of all, Task agent receives jobs from users and distributes according to different user levels to different prioritize global queues to obtain the optimal job distribution based on user levels, AHS uses agent based job distribution strategy at global level. It also handles job priorities at local levels that provide efficient and effective execution of jobs. For different global queues, to assigning jobs to global queue, the priorities are proposed as threshold values. If at any moment, jobs have same priorities then execution of job having minimum run time will take first otherwise First Come First Serve (FCFS) algorithm is used. AHS has optimal performance with respect to QoS parameters.

3. CONCLUSION

Scheduling is one of the major issues in the management of application execution in cloud Scenario. In this paper, based on different parameters like time, cost, and make span, speed, scalability and resource allocation, analysis and overview of various different job scheduling algorithms is carried out. Balance-Reduce algorithm is a data locality driven task scheduling algorithm, which finds a good solution in time $O(\max\{m+n, n \log n\} \cdot m)$. Adaptive Resource Allocation for Pre-emptable Jobs in Cloud Systems algorithm adjusts the resource allocation adaptively based on the updated state of the actual task executions. It works well in intense resource contention situation. The experimental results of Improved cost-based algorithm for task scheduling in Cloud computing show that the time taken to complete tasks after grouping the tasks is very less when compared with time taken to complete the tasks without grouping the tasks. Priority based Weighted Fair Scheduling Algorithm considers certain (especially one of the) parameters of jobs to schedule whereas Priority based Job Scheduling Algorithm considers all the parameters of jobs to perform scheduling. Improving one of the main parameter like makespan, throughput and consistency of the existing algorithm is considered as a future work.

4. REFERENCES

- [1] The NIST definition of cloud computing, NIST special publication 800-145.
- [2] Somassundaram Thamari Selvi and Kannan Govindarajan "CLOUDRB: A framework For Scheduling and managing High-Performance Computing(HPC) applications in science Cloud." Future Generation Computer Systems 34(2014): 47-65
- [3] Thomas A. henzinger, Anmol V.Singh, Vasu Singh. Thomas Wies, "Static Scheduling in Clouds".
- [4] 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.
- [5] M.Arora S.K.Das R.Biswas "A Decentralized Scheduling and Load Balancing Algorithm For heterogeneous Grid Environments", "Proc. Of International Conference on Parallel Processing Workshop (ICPPW'02)", Vancouver, British Columbia Canada, August 2002, pp.400-505
- [6] Amalarethinam, D. I., and Palaniandy Muthulakshmi. "An Overview of the Scheduling Policies and Algorithms in Grid Computing." International Journal of Research & Reviews in Computer Science 2.2 (2011).
- [7] Fatos Xhafa, Ajith Abraham, "Computational models and heuristic methods for Grid scheduling problems", "Future Generation Computer Systems 26", 2010, pp.608-621.
- [8] Yun-Han Lee et al, Improving Job Scheduling Algorithms in a Grid Environment, Future Generation Computer Systems, 27 (2011) 991–998.
- [9] Jiahui Jin, Junzhou Luo, Aibo Song, Fang Dong and Runqun Xiong, BAR: An Efficient Data Locality Driven Task Scheduling Algorithm for Cloud Computing, 2011 11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing.
- [10] Jiayin Li, Meikang Qiu, Jian-Wei Niu, Yu Chen, Zhong Ming Adaptive Resource Allocation for Pre-emptable Jobs in Cloud Systems, 2010 IEEE.
- [11] Mrs S .Selvarani Dr G Sudha Sadhasivm "Improved Cost-Based Algorithm For Task Scheduling In Cloud Computing", 978-1-4244-5967-4/10/\$26.00©2010 IEEE.
- [12] Liu, Xiaocheng, et al. "Priority-Based Consolidation of Parallel Workloads in the Cloud." (2012): 1-1.
- [13] Li Yang et al, A new Class of Priority-based Weighted Fair Scheduling Algorithm, Physics Procedia, 33 (2012) 942 – 948.
- [14] Ghanbari Shamsollah and Mohamed Othman, "A Priority based Job Scheduling Algorithm in Cloud Computing" Procedia Engineering 50(2012): 778-785.
- [15] T L Satty, How to Handle Dependence With the Analytic Hierarchy Process, Math Modeling, 9(3-5) (1987) 369-376.
- [16] T.L.Saaty, Decision Making for Leaders; The Analytical Hierarchy Process for Decisions in a Complex World, Pittsburgh: RWS Publications, (1982,2000).
- [17] Shah, Syed Nasir Mehmmod, et.al "Agent Based Priority Heuristic for Job Scheduling on Computational Grids" Procedia Computer Science 9 (2012):479-488.