

Study of Various Scheduling Algorithm in Cloud Environment

Akash Sharma*, Vinayak Kumar* and Anup Singh Kushwaha**

*Student

**Assistant Professor Computer Science Department

Manav Rachna University

Faridabad, Harayana - 121004, India

Abstract— Cloud Computing is one of the emerging technology based upon on demand pay per use model. It is a platform where various services like applications, bandwidth and data are provided to its users using Internet. The main objective of Job Scheduling Algorithms in Cloud Computing is to optimize the resource allocation and utilization to meet user requirements and for cloud service providers it is the efficient use of resources and thus attaining maximum profit. All this leads us right to the requirement of Job Scheduling in Cloud Computing. Scheduling is the method of deciding how to provide resources amongst the various available tasks or processes so as to achieve the maximum throughput efficiently. In this paper various Job Scheduling algorithms have been presented on the basis of different parameters which provide efficient cloud services.

I. INTRODUCTION

Cloud Computing, one of the prominent technologies present today, is shaping the world of computing by rapid development and deployment of numerous distributed applications. In broad terms, a Cloud could be defined as an elastic execution of resources inside an environment which involves multiple stakeholders and provides us with a metered service at different granularities in respect to a specific level of quality of service. Cloud is a model that gives us a convenient and an available on-demand network access to a shared pool of resources which could be configured and requires minimal computing resources management or service provider interaction.

[1] Cloud computing provides us individuals and organizations with services which have avant-garde software and hardware solutions managed by third party companies in remote locations. A few examples of cloud computing that help in the smooth flowing of our daily lives are like hosting a web application, online file storage or just another social networking site. For the efficient management of the cloud computing environments, Job Scheduling is one of those methods which plays a primary role in generating the best results possible. A decent Job Scheduling algorithm allows a system to perform tasks in a specific order to maximize the throughput. This technology possesses a great potential to match the requirements hungry scientific

computing community. Clouds provide us with a cheap alternative for supercomputers and specialized clusters, it is indeed a much more reliable platform in comparison to grids and is also easily scalable in accordance to the needs. Clouds, when compared to real Supercomputer clusters, furnish a instant scale up which is not the case with Supercomputers [2] that is why it is more important to do proper scheduling and gain the most fruitful and efficient result.

II. TYPES OF CLOUD

On Basis of Deployment Model Cloud is mainly of four types Private, Public, Hybrid and Community.

A. Private Cloud:

A Private Cloud is like a private network which provides and caters its services to a limited number of people. This type of cloud is generally maintained and used within an organization solely for their internal purposes.

B. Public Cloud:

A Public Cloud is the one which sells its services to anyone available on Internet. In this type of cloud, a general user can rent cloud services from cloud providers on dem

C. Hybrid Cloud:

A Hybrid Cloud is a combination of both Private and Public Cloud. This type of cloud is composed of multiple internal or external cloud. This is the scenario when an organization chooses to provide some of the resources internally and outsourcing other resources publically.

D. Community Cloud:

Community Cloud is the combination of Private-Public- Hybrid Cloud.

III. THREE SERVICE MODELS OF CLOUD

Software as a Service(SaaS): SaaS provides the user with custom applications which are built and totally deployed on cloud by web developers. It is used to deploy a single

application through a single web browser to a very large span of customers while using a multitenant architecture on the customer side. SaaS does not require a front-end investment in servers or in the licensing of software in software is responsible for running and maintaining all the essential hardware and software. A customer of SaaS can access all such applications through Internet. Google Docs is a great example of SaaS where a user can perform creation, formatting, deletion, etc. in real-time and also share these documents, spreadsheets or presentations.

- Platform as a Service(PaaS): PaaS provides development of applications and deployment platforms in the form of a service to the web developers. It enables a user to build his/her own applications that could be run on the providers infrastructure and can be used for supporting transactions, uniform authentication, vigorous scalability and availability. The applications built using PaaS are offered as SaaS and are executed directly from the web browsers of the end-users. Thus it provides one with the ability to integrate or consume third-party web services like Google App Engine from other service platforms.
- Infrastructure as a Service(IaaS): IaaS encompasses hardware components of the cloud like server and storage. It allocates the users of the cloud service with greater flexibility at the lower level. In addition, IaaS also gives CPU (Central Processing Unit) clocks along with OS (Operating System) level control to the developers. Amazon EC2 and S3, Virtual Machines, Storage and Load Balancer are some of the examples.

IV. JOB SCHEDULING

Job Scheduling is a mapping mechanism that maps the users tasks to the appropriate selection of resources and their execution. It is a pivotal concept of cloud computing systems task scheduling. The main concern in Job Scheduling revolves around how the efficiency and the throughput of the whole cloud computing system could be increased. Depending upon the business requirements, functions, or priorities, the Jobs and Job Streams can be scheduled to run whenever required. Such processes can be set up daily, weekly, and or even yearly in advance. Inclusive of the above features, these processes can also be set up to run jobs on demand without the requirement of a support staff. [3] The aim of job scheduling algorithms in Cloud Computing systems is to distribute the load among the processors and maximize their utilization while reducing the total task execution time to a minimum. The jobs need to be scheduled to the adaptable resources in respect to adaptable time. This involves the researching of a chronological order in which these jobs could be executed while under transaction logic constraints.

respect to the providers side. As only one application needs to be maintained, the cost is drastically cut down when compared to conventional methods of hosting. Inside SaaS, the seller or the publisher of

The two main types of scheduling algorithms in existence are Static scheduling algorithm and Dynamic scheduling algorithm. Both of these scheduling algorithms have their own advantages and limitations. When compared, the Dynamic scheduling algorithm has a higher performance than the static algorithm but it has a significantly higher overhead cost.

Scheduling Algorithm

In today's world, there has been an implementation of various types of scheduling algorithms in an environment incorporating of distributed computing systems. Most of such algorithms are applied in the cloud environment after suitable verifications. The main advantage of job scheduling algorithms is aimed at achieving a high-performance computing and the best possible system throughput. Conventional job scheduling algorithms are unable to provide scheduling in the cloud environments.

In accordance to a simple classification, Job scheduling algorithms in cloud computing could be categorized in mainly two groups: Batch Mode Heuristic Scheduling Algorithms (BMHA) and Online Mode Heuristic Scheduling Algorithms (OMHA).

- In BMHA, the Jobs are queued and collected into a set on the basis of their arrival in the system. The scheduling algorithm starts after a fixed period of time. The main examples of BMHA based algorithms are: First Come First Serve(FCFS) scheduling algorithm, Round Robin(RR) scheduling algorithm, Min-Min algorithm and Max-Min algorithm.
- In OMHA, Due to the heterogeneity of the cloud environment and the varying speed of each processor, the OMHA are a more appropriate choice. A suitable example is Most Fit Task Scheduling Algorithm (MFTS).

V. SOME FAMOUS JOB SCHEDULING ALGORITHMS

- 1) First Come First Serve (FCFS) In FCFS, the jobs are executed on a first come, first serve basis. It is of two types: non-preemptive or pre-emptive scheduling algorithm. The implementation is based upon a FIFO (First In First Out) queue. To achieve parallel processing, FCFS targets the resource with the smallest waiting queue time and selects it for the incoming task. The VM (Virtual Machine) provisioned component is responsible for the allocation of application-specific VM to hosts in a Cloud-based data center. Due to a high average wait time, FCFS has a poor performance.

Shortest tasks which are at the end of the queue are required to wait for the long task at the front to finish. The turnaround time and response time is considerably low.

- 2) Shortest Job Next (SJN) SJN, also known as shortest job first(SJF), is a non-preemptive scheduling algorithm in which the waiting job or the process with the smallest estimated Burst Time is executed next. Whenever the CPU is available, it is assigned to the process with the next smallest CPU burst. Therefore, it is one of the best approaches to minimize the waiting time. The SJN scheduling is very appropriate for batch jobs where the run times or the execution times are known in advance. Since this algorithm gives the minimum average time for a given set of processes, it probably can be considered as optimal. The SJN algorithm favors short jobs (or processors) at the expense of longer ones. The main problem with SJN scheme is that it needs a precise knowledge of the running time of a job or process and this information is not usually available.
- 3) Round Robin Scheduling (RR) Round Robin (RR) algorithm is focused on the principle of fair sharing. It is a preemptive process scheduling algorithm. Here, each process is given a fixed time to execute, this is called a quantum. Once a process gets executed for the given time period, it is preempted and other process comes into execution for a given time period. Context switching is used to save states of preempted processes. If the time quantum is too large, then it starts behaving like the First Come First Serve scheduling algorithm. RR uses the ring as its queue for storing the jobs. Each job inside a queue has the same amount of execution time and are executed in turn. If a job isn't completed during its turn, it will be stored at the end of the queue and is waited until its next turn. The advantage of RR algorithm is that each job is executed in turn and the jobs don't have to be waited for the previous one to get completed. If the load is found to be heavy, RR takes quite a long time to complete all the jobs. The Cloud Sim toolkit supports RR scheduling strategy for the internal scheduling of jobs. The major drawback of RR is that the largest job takes the most amount of time for completion.
- 4) Priority Algorithm (PA) Priority scheduling Algorithm is a non-preemptive algorithm and it is one of the most common scheduling algorithms among batch systems. In PA, each process is assigned a priority. The process with highest priority is executed at first and the order proceeds according to the priority list. Processes with same priority are executed on a first come first served basis. The priority for a job is decided based upon the

memory requirements, time requirements or any other form of resource requirements. In this strategy, the tasks are initially prioritized according to their size such that the one that has the largest size has the highest rank. The Virtual Machines are also ranked (prioritized) according to their Million Instructions Per Second(MIPS) value such that the one having the highest value of MIPS has the highest rank. Hence, the key factor for prioritizing tasks is their size and for VM is their MIPS. It is a better algorithm than FCFS and RR scheduling.

- 5) Min-Min Algorithm (MM) Min-Min is initiated with a set of unassigned tasks. At first, it computes the minimum completion time for all tasks on all the resources. Then among all these minimum times the minimum value is selected which is actually the minimum time among all the tasks on any of the resources. In the next step, this task is scheduled on the resource on which it takes the minimum time and then the available time of that resource is updated for all the other tasks. After this, the assigned task is not considered and the same process is repeated until each and every task is assigned with the resources.
- 6) Max-Min Algorithm Max-Min is very much similar to the min-min algorithm except for one step. In this algorithm, after finding out the completion time, the minimum execution times are found out individually for all the tasks. Among these minimum times the maximum value is selected which is the maximum time among all the tasks on any resources. Further this task is scheduled on the resource on which would take the minimum time. Later, the available time of that resource is updated for the rest of the tasks. The updating is done in the same fashion as in the Min- Min algorithm. All the tasks are assigned resources by this procedure.
- 7) Most Fit Task Scheduling Algorithm In this algorithm the tasks or processes which fit best in queue are executed first. This algorithm has a high failure ratio.
- 8) Priority Scheduling Algorithm In the Priority Scheduling Algorithm, each process is assigned a priority, and this priority is allowed to run. Equal-Priority processes are scheduled in a FCFS order. The Shortest Job First (SJF) algorithm represents a special case of general priority scheduling algorithm. An SJF algorithm is simply a priority algorithm where the priority is an inverse of the (predicted) next CPU burst,
- 9) i.e. the longer the CPU burst, the lower would be the priority and vice-versa. Priority can be defined either internally or externally. The internally defined priorities use some measurable quantities or qualities to compute the priority of an incoming process.

10) Genetic Algorithm (GA) Genetic algorithm is a problem-solving method that makes the use genetics as a model for problem solving. It is a search-based technique to find out the optimized solution. GA consists of a population of possible solutions. Each solution is represented with a chromosome. It is a method of scheduling in which the tasks are assigned resources in accordance to their individual solutions (which are called schedules in context of scheduling). This us tells about which resource is to be assigned to which particular task. Genetic Algorithm is based upon the biological concept of population generation. Here, a randomized generation of the initial population is done [2].

VI. COMPARISON

The testing of various scheduling algorithms has been done by Savitha. P, J Geetha Reddy [4] that are presently being used in Job Scheduling in Cloud. The following table shows the number of Job scheduling algorithms and a comparative analysis on the basis of their complexity, allocation, waiting time and the type of system. The complexity defines what type of algorithm would be simpler to use in processing.

Allocation defines how the jobs would be assigned to the resources. The Waiting Time defines which algorithm takes more time for processing. The Type of System shows which algorithm is suitable for which particular type of system.

Existing Scheduling Algorithms used on cloud for Job Scheduling

The Following scheduling algorithms are currently prevalent in clouds.

- 1) Resource-Aware-Scheduling algorithm (RASA): Saeed Parsa and Reza Entezari-Maleki [5] displayed a new task scheduling algorithm called RASA. It is a creation of two regular planning Algos: Max-min and Min-min. RASA utilizes the benefits of Max-min and Min-min Algo and beats their detriments. Here, the end of each job, arriving rate of the jobs, cost of its execution on every one of the asset and the cost of correspondence are not considered. The exploratory outcomes demonstrate that RASA beats the current planning Algo in an extensive scale circulated environment.
- 2) RSDC (Reliable Scheduling Distributed in Cloud Computing): Arash Ghorbannia Delavar, Mahdi Javanmard, Mehrdad Barzegar Shabestari and Marjan Khosravi Talebi [6] finds a dependable scheduling algo for the cloud. In this algo, the one task is separated into task. With a specific end goal to keep up a harmony between the tasks, the demand time and acknowledge time are ascertained independently. The scheduling of

task is done by the figuring of the demand time and acknowledge time as a common. That is why, the effectiveness of the cloud is upgraded.

- 3) An Optimal Model for Priority based Service Scheduling Policy for Cloud Computing Environment: Dr. M. Dakshayini, Dr. H. S. Guruprasad [7] proposed a replacement planning algorithmic rule based mostly upon the priority and admission management theme. Here a priority is appointed to every admitted queue. The admission for every queue is finished by calculative tolerable delay and also the service price. The advantage of this algorithmic rule is that this policy in conjunction with the planned cloud design has achieved AN nearly excellent figure (99 percent) of service completion rate with absolute Quality of Service (QoS). It provides the very best precedence for extremely paid user service-requests and thence the service price for the cloud conjointly will increase.
- 4) A Priority based Job Scheduling Algorithm in Cloud Computing: Shamsollah Ghanbari, Mohamed Othman [8] proposed a brand new planning algorithmic program supported multi criteria and multi - call priority driven planning algorithmic program. This planning algorithmic program accommodates 3 level of scheduling: object level, at-tribute level and alternate level. during this algo program priority will be set by job resource quantitative relation. Then priority vector will be compared with every queue. This algorithmic program has higher outturn & fewer end time.
- 5) Extended Max-Min Scheduling Using Petri Net and Load Balancing: El-Sayed T. El-kenawy, Ali Ibraheem El-Desoky, Mohamed F. Al-rahamawy [9] has proposed another algo in light of effect of RASA. Enhanced Max-min algo depends on the normal execution time rather than finish time as a determination premise. Petri nets are utilized to display the simultaneous conduct of circulated frameworks. Max-min exhibits accomplishing plans with practically identical lower make traverse instead of RASA and unique Max-min.
- 6) An Optimistic Differentiated Job Scheduling System for Cloud Computing: Shalmali Ambike, Dipti Bhansali, Jae Kshirsagar, Juhi Bansiwala [10] has proposed a separated algo with non-preemptive priority lining model for jobs performed by cloud client in the distributed cloud. In this approach 1 web app is made to do few task like one of the record up- stacking and downloading at that point there is need of productive job planning algo. The QoS

necessities of the cloud client and the greatest benefits of the cloud are accomplished with this algo.

- 7) Improved Cost-Based Algorithm for Task Scheduling: Mrs.S.Selvarani, Dr.G.Sudha Sadhasivam [11] thought of an enhanced cost-based scheduling algo to play out an effective mapping of errands to the accessible assets in the cloud. The spontaneous creation of an ordinary action based costing is proposed by another errand planning system for the cloud where there are no relations between the overhead application base and in the manner by which different undertakings cause overhead cost of assets in cloud. This planning algo separates every client undertaking relying upon its need of into three distinct records. It gauges both the asset cost and calculation execution and furthermore enhances the calculation correspondence proportion.
- 8) Performance and Cost evaluation of Gang Scheduling in a Cloud Computing System with Job Migrations and Starvation Handling: Ioannis A. Moschakis and Helen D. Karatza have proposed a pack arranging estimation with work development and starvation dealing with which incorporates the booking of parallel occupations which are starting at now associated in the zones of Grid and Cluster enrolling. The amount of VMs available at any event is dynamic and is scaled by the solicitations of the businesses being updated. This model is considered through reenactment and empowers one to analyze the execution and the general cost. Results have shown that this booking method can be satisfactorily passed on Clouds and such cloud stages are sensible for High Performance Computing or world class wander applications.

TABLE I
 COMPARISON OF VARIOUS TRADITIONAL
 SCHEDULING ALGORITHMS ON DIFFERENT
 PARAMETERS

Algorithms	Complexity	Allocation	Waiting Time	Type of system
First Come First Serve Algorithm(FCFS)	Simplest Scheduling Algorithm	CPU is allocated in the order in which the processes arrive	Average Waiting Time is High	Batch system
Shortest Job First Algorithm(SJF)	Overhead of Calculating Burst Time of Processes	CPU is allocated to the process with least CPU burst time	Average Waiting Time is less	Batch system
Priority Algorithm	Difficulty in preventing Starvation	Based on priority, So the higher priority job can run first.	Waiting Time is Less for Higher Priority Processes	Batch and Time Sharing Systems
Round Robin	Performance heavily depends upon	The preemption takes place	Average Waiting Time is	Time Sharing System

Algorithm(RR)	the size of time quantum	after a fixed interval of time called Time Quantum	Less	
Genetic Algorithm	Complexity depends on the task to be scheduled	This is a greedy algorithm and picks the best job/process to allocate the CPU	Waiting Time is less	Batch and Time Sharing Systems

VII. CONCLUSION

Job Scheduling is one of the most important task in cloud computing environment. In this paper we have analyzed various Job scheduling algorithms and tabulated various parameters. We have noticed that existing scheduling algorithms gives high throughput and are cost effective but they do not consider reliability and availability. So we need algorithms that can improve availability and reliability in cloud computing environment.

REFERENCES

- [1] National Institute of Standards and Technology - Computer Security Resource Center - www.csrc.nist.gov
- [2] Iosup, Alexandru, et al. "Performance analysis of cloud computing services for many-tasks scientific computing." IEEE Transactions on Parallel and Distributed systems 22.6 (2011): 931-945.
- [3] Hong Sun, Shi-ping Chen, Chen Jin, and Kai Guo1, "Research and Simulation of Task Scheduling Algorithm in Cloud Computing", Telkommika, Vol.11, No.11, November 2013, pp. 6664-6672
- [4] Reddy, J. Geetha. "A review work on task scheduling in cloud computing using genetic algorithm." International Journal of Scientific and Technology Research 2.8 (2013): 241-245.
- [5] Parsa, Saeed, and Reza Entezari-Maleki. "RASA: A new task scheduling algorithm in grid environment." World Applied sciences journal 7.Special issue of Computer and IT (2009): 152-160.
- [6] Arash Ghorbannia Delavar, Mahdi Javanmard, Mehrdad Barzegar Shabestari and Marjan Khosravi Talebi RSDC (RELIABLE SCHEDULING DISTRIBUTED IN CLOUD COMPUTING) in International Journal of Computer Science, Engineering and Applications (IJCEA) Vol.2, No.3, June 2012
- [7] 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) Volume 32 No.9, October 2011
- [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)
- [9] El-Sayed T. El-kenawy, Ali Ibraheem El-Desoky, Mohamed F. Al-rahmawy Extended Max- Min Scheduling Using Petri Net and Load Balancing International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-4, September 2012
- [10] Shalmali Ambike, Dipti Bhansali, Jae Kshirsagar, Juhi Bansiwala An Optimistic Differentiated Job Scheduling System for Cloud Computing International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 2, Mar-Apr 2012, pp.1212-1214
- [11] Mrs.S.Selvarani, Dr.G.Sudha Sadhasivam, improved cost-based algorithm for task scheduling in Cloud computing ,IEEE 2010.