Queue based Job Scheduling algorithm for Cloud computing

Shahram Behzad¹, Reza Fotohi², Mehdi Effatparvar²

1. Department of Computer Engineering, Germi branch, Islamic Azad University, Germi, Iran
2. ECE Department, Ardabil Branch, Islamic Azad University, Ardabil, Iran

*Corresponding Author email: sh.behzad173@gmail.com

ABSTRACT: Cloud Computing is a revolutionary way of providing shared resources over the Internet. Through the use of low level virtualization software, the Cloud provides virtualized computing hardware infrastructure in a manner similar to the public utilities, thus it is also termed as Infrastructure-as-a-Service or Utility Computing. Job scheduling algorithms is one of the most challenging theoretical issues in the cloud computing area. Some intensive researches have been done in the area of job scheduling of cloud computing. This paper presents a queue based job scheduling algorithm for efficient execution of user jobs. This paper also includes the comparative performance analysis of our proposed job scheduling algorithm along with other well-known job scheduling algorithms considering the parameters like average waiting time, average response time. The result has shown also exhibit that Our proposed scheduling algorithms (QHS ) has shown the best average waiting times, average response times compared to other job scheduling approaches

Keywords: Cloud Computing, Job Scheduling, Distributed Systems, AHP.

INTRODUCTION

Scheduling strategies have been studied before under the notion of Cloud Computing. (Assunção MD et al., 2010). Asunció et al. studied the use of CC as an extension to private clusters.

In their model, tasks were separate from each other and did not communicate. Virtual Machine usage and leasing has also been studied (Blazewicz et al., 2001). through the use of Haizea1 VM-based lease management architecture. Several job scheduling algorithms have been proposed in distributed computing area (Mousumi et al., 2011). (Silberschatz et al., 2011). And (Sotomayor et al., 2008). Most of them can be applied in the cloud environment with suitable verifications (Mousumi et al., 2011). (Madhooshi et al., 2007). (Mousumi et al., 2001). The main goal of job scheduling is to achieve a high performance computing and the best system throughput. Traditional job scheduling algorithms are not able to provide scheduling in the cloud environments. According to a simple classification (Li Y et al., 2011). job scheduling algorithms in cloud computing can be categorized into two main groups; Batch mode heuristic scheduling algorithms (BMHA) and online mode heuristic algorithms. In BMHA, Jobs are queued and collected into a set when they arrive in the system. The scheduling algorithm will start after a fixed period of time. The main examples of BMHA based algorithms are; First Come First Served scheduling algorithm (FCFS), Round Robin scheduling algorithm (RR), Min–Min algorithm and Max–Min algorithm. By On-line mode heuristic scheduling algorithm, Jobs are scheduled when they arrive in the system. Since the cloud environment is a heterogeneous system and the speed of each processor varies quickly, the on-line mode heuristic scheduling algorithms are more appropriate for a cloud environment. Most fit task scheduling algorithm (MFTF) is suitable example of On-line mode heuristic scheduling algorithm (Li Y et al., 2012).

Suitable job scheduling algorithm must consider priority of jobs. To address this problem some researchers have considered priority of jobs scheduling algorithm (Monir Abdullah et al., 2010). (Navendu J et al., 2011). (Shokripour A J et al., 2011). Those researches have focused on a few criteria of jobs in scheduling. In cloud environments we always face a wide variety of attributes that should be considered. It means a particular job scheduling algorithm in cloud environments should pay attention to multi-attribute and multi-criteria properties of jobs. AHP also is a suitable method for priority based problems such as scheduling with various attributes and alternatives (K.S. Chatrapati et al., 2010). And (Silberschatz et al., 2001).

This paper also includes the comparative performance analysis of our proposed job scheduling algorithm along with other well-known job scheduling algorithms considering the parameters like average waiting time, average response time.
The remainder of this paper is organized as follows: Section 2 describes the proposed queue based hybrid scheduling algorithm. Section 3 presents the performance evaluation of cloud scheduling algorithms. Conclusion is given in the final section.

Proposed Queue based Hybrid Scheduling Algorithm

In this method, a central node considered as Job queue plays the role of taking jobs from users and storing them in its queue. Then, the jobs are ordered based on priority, and thereafter, the jobs are numbered from 1 to n for each priority. Then, the time quantum of job 1 to job n is calculated in parallel. Then, the user with a lower quantum time is CPU allocated and executed.

Method of Time Quantum Calculation

According to the below relation, time quantum is calculated based on the average of CPU burst time for user 1 to user n in parallel. Then, the user with a lower time quantum compared to other users is CPU allocated and executed.

\[ Q_{User \, 1} = \text{Average} (JCT_1, JCT_2, JCT_3, \ldots, JCT_i) \]
\[ Q_{User \, 2} = \text{Average} (JCT_{i+1}, JCT_{i+2}, JCT_{i+3}, \ldots, JCT_j) \]
\[ Q_{User \, 3} = \text{Average} (JCT_{j+1}, JCT_{j+2}, JCT_{j+3}, \ldots, JCT_n) \]

In the above relation, the JCT variable represents job processing time. In this proposed method, due to the fact that we queued the jobs and that time quantum is calculated in parallel for all users, hence, this method results in improved waiting time and response time, and acts better than FCFS, RR and SJF scheduling algorithms. JCT (Job CPU TIME)

Figure 1. Block Diagram of Queue based Hybrid Scheduling (QHS)

Scheduling Algorithms

Job scheduling algorithms is one of the most challenging theoretical issues in the cloud computing area. Some intensive researches have been done in the area of job scheduling of cloud computing. Jobs are queued and collected into a set when they arrive in the batch mode. The scheduling algorithm will start after a fixed period time.
First come first service scheduling algorithm (FCFS)

In this algorithm, jobs are executed according to the order of job arriving time. The next job will be executed in turn. The FCFS algorithm (Shokripour et al., 2012), may induce a “convoy effect”. The convoy effect happens when there is a job with a large amount of workload in the job queue. When this occurs, all the jobs queued behind it must wait a long time for the long job to finish.

Shortest job first scheduling algorithm (SJF)

A different approach to CPU scheduling is the shortest-job-first (SJF) scheduling algorithm. This algorithm associates with each process the length of the process's next CPU burst. When the CPU is available, it is assigned to the process that has the smallest next CPU burst. If the next CPU bursts of two processes are the same, FCFS scheduling is used. The SJF algorithm can be either pre-emptive or non-preemptive. The choice arises when a new process arrives at the ready queue while a previous process is still executing. The next CPU burst of the newly arrived process may be shorter than what is left of the currently executing process. A pre-emptive SJF algorithm will preempt the currently executing process, whereas a non-preemptive SJF algorithm will allow the currently running process to finish its CPU burst. Pre-emptive SJF scheduling is sometimes called shortest-remaining-time-first scheduling.

Round Robin scheduling algorithm (RR)

The RR algorithm (Sunita B et al., 2001), mainly focuses on the fairness problem. The RR algorithm defines a ring as its queue and also defines a fixed time quantum. Each job can be executed only within this quantum, and in turn. If the job cannot be completed in one quantum, it will return to the queue and wait for the next round. The major advantage of RR algorithm is that jobs are executed in turn and do not need to wait for the previous job completion. Therefore, it does not suffer from a starvation problem. However, if the job queue is fully loaded or workload is heavy, it will take a lot of time to complete all the jobs. Furthermore, a suitable time quantum is difficult to decide.

Performance Metrics

Performance metrics for the cloud scheduling algorithms are based on three factors - average waiting time, average response time. We performed experiments for different scheduling algorithms (Shokripour B et al., 2012). We formed two data sets by using workload i.e. 7000 and 15000 processes. We performed an experiment by varying the number of CPUs from 4 to 64. We used ‘20’ units as the fixed time quantum for our experiment.

In this section, we describe a comparative performance analysis of our proposed algorithms, i.e. QHS, with three other Cloud scheduling algorithms; i.e. FCFS, SJF, and RR.

Average waiting time

Waiting time is defined as how long each process has to wait before it gets its time slice. In scheduling algorithms such as Shorted Job First and First Come First Serve, we can find that waiting time easily when we just queue up the jobs and see how long each one had to wait before it got serviced. When it comes to Round Robin or any other preemptive algorithms, we find that long running jobs spend a little time in CPU, when they are preempted and then wait for some time for its turn to execute and at some point in its turn, it executes till completion.

Average response time

It is the amount of time taken from when a process is submitted until the first response is produced (Yun-Han et al., 2001). (Sotomayor B et al., 2009). Average response times for each algorithm have decreased by increasing the number of CPUs.

Performance Evaluation of Cloud Scheduling Algorithms

Performance metrics for the cloud scheduling algorithms are based on two factors - Average Waiting Time, Average Response Time. We performed experiments for different scheduling algorithms (Sotomayor et al., 2009). We formed two data sets by using workload i.e. 7000 and 15000 processes. We performed an experiment by varying the number of CPUs from 4 to 64. We used ‘20’ units as the fixed time quantum for our experiment. In this section, we describe a comparative performance analysis of our proposed algorithms, i.e. QHS, with three other cloud scheduling algorithms; i.e. FCFS, SJF and RR.

Average Waiting Times Evaluation
The Waiting Time is the time for which a process waits from its submission to completion in the local and global queues (Yun-Han et al., 2001). Fig. 2 and Fig. 3 shows that the average waiting times computed by each scheduling algorithm for each real workload trace of 7000 and 15000 processes. That the SJF and QHS scheduling algorithms produce the shortest average waiting times as compared to the other scheduling algorithms. By increasing the number of CPUs, each algorithm shows the relative improvement in performance. Also, the FCFS and RR have shown the worst performance the average waiting time measures. As a result, CHS has shown the optimal average waiting times for 7000 and 15000 processes.

Average Waiting Time Evaluation

Average response times for each algorithm have decreased by increasing the number of CPUs. Fig. 4 and Fig. 5 shows that the average response times computed by each scheduling algorithm for each real workload trace of 7000 and 15000 processes. The FCFS and SJF scheduling algorithms result in poor response times as compared to the other scheduling algorithms. It also shows that RR and QHS algorithms produces better average response time compared to other algorithms.
CONCLUSION

In this paper, a queue based scheduling algorithm of the cloud computing is proposed, namely QHS. We compared the performance of proposed job scheduling algorithm with other grid scheduling algorithms on a computing cloud. Simulation results show that QHS has shown the optimal performance in terms of average waiting times and average response time. Simulation results also exhibit that RR has shown the best average response times compared to other job scheduling approaches.


Shahram Behzad received his B.Sc. in Computer Engineering from Parsabad University, Parsabad, Iran, in 2008, and his M.Sc. in Computer Engineering from Islamic Azad University, Germ branch, Ardabil, Iran, in 2013. His research interests include Mobile Ad-hoc Networks, Wireless Sensor Networks, Distributed Systems, Performance Evaluation, and Optimization Algorithms.

Reza Fotohi received his B.Sc. in computer engineering from Shabestar University of Applied Science And Technology, Tabriz, Iran, in 2009, and his M.Sc. in Computer Engineering from Islamic Azad University, Germi branch, Ardabil, Iran, in 2013. His research interests include Mobile Ad-hoc Networks, Wireless Sensor Networks, Distributed Systems, Performance Evaluation, and Optimization Algorithms.

Mehdi Effatparvar is faculty member of computer engineering department in Islamic Azad University of Ardabil, Iran. He is PhD student in Islamic Azad University of Science and Research. He received his BSc in Computer engineering and MSc in Information Technology from Islamic Azad University of Qazvin, Iran. His research interests include wireless sensor networks, ad-hoc networks, distributed systems and operating systems.