Aggressive Migration: An Effective Scheduling Policy

Manjeet Kaur, Sushil Kamboj

Shaheed Udham Singh College of Engineering & Technology, Tangori

Abstract: *Cloud computing is a terminology which is achieved by Distributed and Parallel computing. It offers end consumers an —a pay as go model and provides services to remote users. To provide easy customer access, many cloud computing companies integrate frameworks for parallel data processing in their product portfolio. Parallel processes require communication and synchronization among various processes. As a result the CPU utilization of parallel applications of certain nature often decreases as parallelism grows. It is challenging for a data center to achieve a certain level of utilization of its nodes while maintaining the level of responsiveness of parallel jobs. Normally, responsiveness is on the top priority for existing scheduling algorithms. In this paper, we review and classify scheduling algorithms in literature.*

Keywords: Cloud computing, Parallel computing, Parallelism, Data center, CPU utilization

1. Introduction

1.1 Introduction to Cloud Computing

According to NIST definition, Cloud Computing is a model for 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 [12].

Job scheduling is one of the major and challenging issues in cloud computing. A good scheduling policy helps in proper and efficient utilization of Virtual Machine's (VMs). Job scheduling algorithm is an NP-complete problem which places an important role in cloud computing [7].

The cloud computing architecture has three layers, for the software which require on demand services over Internet [7], these are:

1) Infrastructure as a Service (IaaS): This layer delivers hardware components (like server and storage) and software as services.
2) Platform as a Service (PaaS): Cloud application developers are the users of this layer. Automatic scaling, load balancing and integration with other services (e.g. email services) are the major benefits to cloud application developer.
3) Software as a Service (SaaS): This layer hosts the software and provide to the customer through Internet. It reduces the purchase and maintenance cost of the customer.

1.2 Schedulers for Clouds

Cloud computing evolved from grid computing, service oriented computing, and virtualization paradigms. This means that scheduling algorithms developed for these types of systems can also be used in clouds. Scheduling algorithms can be distinguished by their main characteristics [9], such as:

1) Target system: The system for which the scheduling algorithm was developed, which can be a heterogeneous system, a grid, or a cloud computing system.
2) Optimization criterion: Makespan and cost are the main metrics specified by cloud user and considered by schedulers in the decision making process.
3) Multicore awareness: Computer systems can have multiple cores, which should be considered by scheduling algorithms in resource selection.
4) On-demand resources: Resources can be leased either on-demand or for long terms. The on-demand leasing of resources is treated by the scheduling algorithm as a "single expense" during the execution of the workflow.
5) Reserved resources: The algorithm should consider the use of a resource reserved for a long term.
6) Levels in a service level agreement (SLA): The scheduling algorithm should consider that SLAs can be organized hierarchically. SLAs with a single level allow clients and providers to interact directly to negotiate resource capacities and prices.
7) Heuristics information: Heuristics information is some problem based values to guide the search direction. Based on the problem of workflow we design heuristics here. The main heuristics are Cost, Makespan, Bandwidth and Multicore.

1.3 Parallel job Scheduling Algorithms

The most basic but popular batch scheduling algorithm for parallel jobs is first come first serve (FCFS). FCFS may cause node fragmentation and to solve this issue backfilling and Gang scheduling methods are introduced [20].

1) Gang Scheduling: Gang scheduling allows resource sharing among multiple parallel jobs. The gang scheduling algorithm manages to make all the processes of a job progress together so that one process will not be in sleep state when another process needs to communicate with it.
2) Backfilling: It allows small jobs to move ahead and run on processors that would otherwise remain idle. The algorithms based on backfilling are as follows:
   a. CMBF (Conservative Migration supported Back Filling): The algorithm assumes that the state of a job

can be saved and restored; therefore, the scheduler is able to suspend a job and resume it on other nodes in a later time.

b) AMBF (Aggressive Migration Supported Back Filling): AMBF only tracks backfilling jobs for the job at the head of the queue and allows the head-of-queue job to preempt other jobs.

c) CMCBF (Conservative Migration and Consolidation supported Back Filling): It ensures that a job is dispatched to run in foreground VMs whenever the number of foreground VMs that are either idle or occupied by jobs arriving later than it satisfies its node requirement.

d) AMCBF (Aggressive Migration and Consolidation supported Back Filling): AMCBF uses the CPU usage information of parallel processes to make scheduling decisions. In AMCBF, there is a delay in the execution of jobs other than first job in the queue. This algorithm challenging to achieve responsiveness of parallel jobs and high processor utilization in the cloud.

2. Related Work

Various Studies have been done on the Workflow scheduling in Cloud. Different Studies showed different results for workflow applications. N. Doan Man et al. discussed that the extension of the local computing platforms required a huge investment of both finance and human power. In this work, a framework built from the combination between the computing resources on Cloud computing and the computing components in the local systems, was presented [13]. L. Zhou et al. discussed that the scheduling of massive multimedia flows with heterogeneous QoS guarantees became an important issue for the mobile cloud. The highlight of this article lied in developing a blind online scheduling algorithm (BOSA). This algorithm assigned available multimedia servers based on the last time-slot information of the users' requests, and route all the multimedia flows according to the first-come-first-served rule [11]. R. Santhosh et al. discussed a scheduling approach to focus on providing a solution for online scheduling problem of real-time tasks using "Infrastructure as a Service" model offered by cloud computing. This scheduling method sensibly aborted the task when it missed its deadline and affected the overall system performance and response time of a task [14]. S. Min Jung et al. discussed that there was no consideration of data priority in Cloudsim. Therefore, it was needed to research a scheduling algorithm to support priority of Cloudsim. The user code and Cloudsim layer were revised and defined the way to support scheduling algorithms in Cloudsim [16]. C. Chi Huang et al. presented a cloud computing based scheduling system using optimized layout method. This system used the structure of SQL Azure as the cloud database for scattering and storing data on disk arrays [3]. D. Ding et al. presented an Adaptive Resource Scheduling Mechanism based on User Behavior Feedback (ARSM-UBF). ARSM-UBF gave attention to the efficiency of cloud system [6]. T. Yi Chen et al. provided early deadline first until zero laxity (EDZL) scheduling algorithm to optimize the cloud service scheduler for ensuring sensor jobs do not miss deadline in the LCPS. EDZL integrated EDF and LLA, and had a low context

switching overhead and a low deadline miss ratio [17]. W. Chen et al. provided a set-based PSO (S-PSO) approach for scheduling. In addition, the S-PSO provided an effective way to take advantage of problem-based heuristics to further accelerate search [19]. Y. Hu et al. discussed that in virtual machine system, competing for CPU resource between different workloads would cause serious performance interference and thus degraded the system performance and efficiency. In order to address this issue, the possibility of scheduling virtual machine asymmetrically based on workload classification and multi-core partitioning, was introduced [22]. B. Yang et al. introduced the job scheduling scenario in the Cloud Computing and its existing solutions. There was no well-defined job scheduling algorithm for the cloud under overloading circumstances and existing algorithms did not take hardware/software failure and recovery in the cloud into account, therefore to address these challenges a Reinforcement Learning framework was introduced which would consider long-term optimization of the scheduling algorithm and tend to be fault-aware as well as productive [2].

3. Proposed Methodology

The methodology includes the implementation of Aggressive Migration and Consolidation Supported Backfilling (AMCBF) algorithm that allows inaccurate CPU usage estimation of parallel processes. Following sequence would be followed during the implementation:

1. To study the performance of AMCBF algorithm and record the observations.
2. To develop a simulated cloud environment for scheduler.
3. To propose a new and improved algorithm for job scheduling.
4. To compare and evaluate the performance of AMCBF and proposed algorithm based on metrics (waiting time, response time, turnaround time).
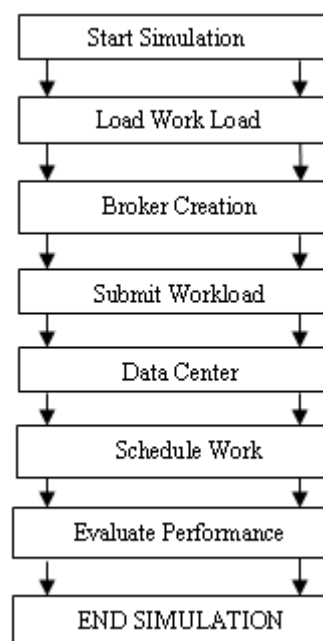
3.1 Flow graph

Figure 1: Flow of scheduling

Paper ID: SEP1460

147

4. Conclusion

In cloud environment, Scheduling is considered as the major factor for task execution. In this paper, we have surveyed the various parallel job scheduling algorithms based on backfilling. The existing scheduling algorithm can be used to introduce more efficient and better improved performance of algorithms in terms of arrival time, execution time of task on resources and cost of the communication. So there is a need to improve availability and reliability of cloud computing for task allocation.

5. Acknowledgement

References

[1] Anju Bala, Inderveer Chana," A Survey of Various Workflow Scheduling Algorithms in Cloud Environment", International Journal of Computer Applications, pp. 26-30, 2011.

[2] Bo Yang, Xiaofei Xu, Feng Tan, Dong Ho Park, " An Utility-Based Job Scheduling Algorithm for Cloud Computing Considering Reliability Factor", IEEE International Conference on Cloud and Service Computing, pp. 95-102, 2011.

[3] Chung-Chi Huang, Chung-Lin Huang," Development of Cloud Based Scheduling System Using Optimized Layout Method for Manufacturing Quality", IEEE International Symposium on Computer, Consumer and Control, pp. 444-447, 2012.

[4] Chunming Liu, Xin Xu, Dewen Hu," Multi-objective Reinforcement Learning: A comprehensive Overview", IEEE Transactions on Cybernetics, pp. 1-13, 2013.

[5] Dhaval Limbani, Bhavesh Oza," A Proposed Service Broker Strategy in Cloud Analyst for Cost-Effective Data Center Selection", International Journal of Engineering Research and Applications, pp.793-797, Jan-Feb 2012.

[6] Dind Ding, Yidong Li, Lihua Ai, Siwei Luo," An Adaptive Resource Scheduling Mechanism based on Behavior Feedback in Cloud Computing", IEEE International Conference on Parallel and Distributed Computing, Applications and Technologies, pp. 543-547, 2012.

[7] Fazel Mohammadi , Dr. Shahram Jamali , Masoud Bekravi," Survey on Job Scheduling algorithms in Cloud Computing", International Journal of Emerging Trends & Technology in Computer Science, pp. 151-154, March – April 2014.

[8] Gaurang Patel, Rutvik Mehta," A Survey on Various Task Scheduling Algorithm in Cloud Computing", International Journal of Advanced Research in Computer Engineering & Technology, pp. 715-717, 2014.

[9] J. Elayaraja , S.Dhanasekar," A Survey on Workflow Scheduling in Cloud using Ant Colony Optimization", International Journal of Computer Science and Mobile Computing, pp. 39-44, February- 2014.

[10] Komal Singh Patel, A.K.Sarje," VM Provisioning Method to Improve the Profit and SLA Violation of Cloud Service Providers", IEEE, 2012.

[11] Liang Zhou, Zhen Yang," Exploring Blind Online Scheduling for Mobile Cloud Multimedia Services", IEEE, June 2013.

[12] Mell, P. and Grance T. (2011), "NIST Special Publication 800-145 : The NIST Definition of Cloud Computing", Retrieved from NIST Information Technology Laboratory,http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf

[13] Nguyen Doan Man, Eui-Nam Huh," Cost and Efficiency-based Scheduling on a General Framework Combining between Cloud Computing and Local Thick Clients", IEEE, pp. 258-263, 2013.

[14] R. Santosh, T. Ravichandrn," Pre-emptive Scheduling of Online Real Time Services with Task Migration for Cloud Computing", IEEE, 2013.

[15] Shahabanath K K, Sreekesh Namboodiri T," Parallel Workload Scheduling in Cloud: A Survey", International Journal of Computing and Technology, pp. 30-33.

[16] Sung-Min Jung, Num-Uk Kim, Tai-Myoung Chung," Applying Scheduling Algorithms with QOS in the Cloud Computing", IEEE, 2013.

[17] Tseng-Yi Chen, Hsin-Wen Wei, Jenq-Shiou Leu, Wei-Kuan Shih," EDZL Scheduling for Large-Scale Cyber Service on Real-Time Cloud",2012.

[18] Wang Han, Tang Xiao-qi," Dynamic Task-Scheduling Algorithm in CNC Based on Cloud Computing", IEEE, 2012.

[19] Wei-Neng Chen, Jun Zhang," A Set-Based Discrete PSO for Cloud Workflow Scheduling with User-Defined QOS Constraints", IEEE, pp. 773-778, 2012.

[20] Xiaocheng Liu, Chen Wang, Bing Bing Zhou, Junliang Chen, Ting Yang, Albert Y. Zomaya," Prority-Based Consolidation of Parallel Workloads in the Cloud", IEEE, 2013.

[21] Yanyan Hu, Xiang Long, Jiong Zhang," Enhance Virtualized HPC System Based on I/O Behavior Perception and Asymmetric Scheduling", IEEE, pp. 169-178,2012.

[22] Yanyan Hu, Xiang Long, Chengjian Wen," Asymmetric Virtual Machine Scheduling Model Based on Workload Classification", IEEE, 2012.

Paper ID: SEP1460

148