

An Efficient Hierarchical Job Scheduling Algorithm for Cloud Computing

AV.Karthick ¹, K.Muthu Pandi ², R.Ganapathy Subramanian ³

Department of International Business, Alagappa University, Karaikudi, India ¹

Department of Commerce, Sourashtra College, Madurai, India ²

Department of Commerce CA, Mannar Thirumalai Naicker College, Madurai, India ³

ABSTRACT: Cloud Computing is one the fastest growing field in Computer Science and Information Technology. The main reason for the growth of Cloud Computing is familiar due to zero maintenance of hardware, software and networking. It is a combination of grid, parallel and distributed computing. The widespread availability and pay per use are added more advantage to cloud computing. Scheduling is a predefined procedure that control over all job process in the cloud system. Scheduling aims at maximize the cloud server resources in a balanced way. The prime objective of scheduling is allocates the workload in a diversified manner of variety of cloudlet. The biggest challenge in cloud computing is client submitting jobs are come from heterogeneous types and vibrant one. Poor resource allocation scheduling policy leads to idle processor. Based on this problem, the proposed hierarchical job scheduling algorithm help to allocates the jobs in a load balanced manner. It helps to attain the proper usage of resources, providing a quality system.

KEYWORDS: Scheduling, Load Balancing, Cloudlet, Hierarchical, Processor

I. INTRODUCTION

Cloud computing is a web based computing that provides different services like software, hardware, infrastructure, desktop and storage as a service. The biggest strength of cloud computing was the user collecting the fees based on their usage called as pay per use. The cloud user need not update the resources like network, hardware and software. Client demand was dynamic one that cannot easily and accurately predict them. Demand generally refers to willingness to buy and ability to pay for the product. Demand plays an important element for planning and decision making. It decides the amount of units to be produced. There was an inverse relationship between demand and price. If the price of the service will be increased the demand will be less. If the demand of the service has increased the price of the service will be reduced. It shows the price increases the demand for access the cloud will be reduced. The factors affect the demand includes the income of the client, quality of service, price of the service provider, Competitor price (substitute), preferences, currency value, offers and advertisement. Scheduling based on Tri queue and multi queue job scheduling leads to better result published by AV.Karthick et al. (2013, 2014, pp.164).

Cloud computing was a recent market offers said by Martin Randles and David Lamb (2010, pp. 551). It projected to enable computing transversely widespread and varied resources, slightly differ from local machines or at remote server farms. Cost provisioning technique for Cloud Computing studies by AV.Karthick et al. (2016, pp.1). Sudha Sadhasivam, R. Jayarani. (2009, pp.585) developed on time delivery of reliable, fault-tolerant and scalable transportation for hosting web based application services. The compile time of mixed cloud based task duplication policy to accomplish the fine grained purpose characterize as directed acyclic graph (DAG) into the mixed cloud atmosphere. It list the jobs based on a degree relative to the critical path in the DAG and try to attain lower bound of the DAG. Job scheduling model given by Luqun Li (2009, pp. 295) system problem was core and complex issue in Cloud Computing. They did not make use of the Cloud computing resource competently and achieve the maximum gain with job scheduling structure was one of the Cloud computing service providers' ultimate goals. Job scheduling system was accountable to pick the preeminent suitable resources in a Cloud by taking a few static and dynamic parameters restrictions for consideration. S.V.Sudha et al (pp.1) find that load imbalance causes harm to performance of the whole parallel application because each thread of computation needs a diverse amount of time to comprehensive

and the entire program must wait for the slowest thread before it can synchronize. Loganayagi B, and S.Sujatha (2010, pp.1) analyzed software industry was change every day. What you get/invest for your company/need nowadays was going to turn into outdated within a decade itself. Upgrading your hardware/ software requirement was also easy with the cloud, lacking disturbing the present work. Scalability and continuance was easy in the case of cloud. Scheduling policy has important impact on the performance individuality of a parallel system. In the early hours policy stated by S.V.Sudha et al (pp.1) used a space sharing approach, in which jobs run side by side on diversified nodes of the machine at the identical time, but each node was exclusively assigned to a job.

II. REVIEW OF LITERATURE

There are numerous number of job scheduling was designed by the researcher. They did not meet the user quality satisfaction in terms of zero defects. They also aims at minimize the systems idle time and also balancing the system load maintenance. Dynamic scheduling cloud resources according to the change of the load are input to enhance cloud computing, on-demand based service potential. This paper load-adaptive cloud resource scheduling model stands on ant colony algorithm. Dynamic resource scheduling was one of the problems to be resolved in a lot of technical fields.

For example, in the field of the database, Brown et al studied self-management of target-oriented disk store, memory management and in the communications field, Xue Qiang et al proposed a load adaptive time slot PQ (priority queue algorithm) to solve the conflict of the real-time traffic Typically tasks are programmed by user requirements. New scheduling policy requires to be projected to overcome the problems generated by network properties sandwiched between client and resources. New scheduling strategies defined by S.Selvarani and G.Sudha Sadhasivam (2010, pp.1) use some of the conventional scheduling concepts to combine them together with some network aware strategies to offer answer for enhanced and more capable job scheduling. A side-effect of this approach was that overall computer usage rises dramatically, as customers do not have to persuade for hit the highest point load limits. In accumulation, enhanced immediate bandwidth creates it credible to accept the equal response times from federal infrastructure at further sites.

Scheduling befall a challenging task, especially when parallel jobs are consider. The First Come First Served (FCFS) algorithm analyzed by Sofia K. Dimitriadou (2010, pp. 329) has confirmed deficient and reason for harsh fragmentation when resources are not accessible for big gangs. In combinational backfill algorithm (CBA) little jobs are getting towering priority. Suresh.A and Vijayakarhick.P (2011, pp. 624) developed improved backfill algorithm used balanced spiral (BS) method for achieve QOS in cloud environment. Several heuristics studied by Sunita Bansal et al. (2011, pp. 472) has proposed to schedule the tasks competently, there was a large amount suitable node there in the grid. A preponderance of these heuristics explains satisfactory outcome in static grid environments. On the other hand, they cannot be straight applied in vibrant environments where tasks are continuously arriving in the grid at regular intervals. In Minimum-minimum scheduling algorithm, every job for all time allocate to the reserve which can entire it the initial in order to spend less time completing all jobs.

The Max-min scheduling algorithm was related to Min-min scheduling algorithm excluding that it supply the highest precedence to the job with the utmost most primitive completion time. Space sharing examined by S.V.Sudha et al (pp.1) in isolation resulted in deprived consumption, nodes are unutilized in spite of ahead of queue in jobs. Also the wait and response time for jobs with an exclusively space sharing strategy can be relatively high. The scheduling of bag-of-tasks J. Octavio et al. (2011, pp.416) applications (BoTs) in Clouds deal with fixed predefined allocation slots of Cloud resources, Cloud property for carry out BoTs are with dynamism composed by accept the agreement net protocol.

Parallel Genetic Algorithm Zhongni Zheng et al. (2011, pp. 444) was much faster than conventional Genetic Algorithm. The research shows that our method improved both the speed of resources allocation and the utilization of system resource. Alexander M. Lindsay et al (2011, pp. 142) analyzed the EASY and Conservative Backfilling used First-Come-First-Serve (FCFS) order. Perkovic et al. deliberated the Conservative Backfilling with random queue ordering by their length and random rearrange as well. EASY backfill algorithm was proposed by Tsafirir et al. Talby et al. combine three types of priorities in the order of queue jobs.

Co-scheduling algorithms model was done by Eitan Frachtenberg, Fabrizio Petrini (2005, pp.1066) such as Implicit Co-Scheduling (ICS), Dynamic Co-Scheduling (DCS), or Coordinated Co-Scheduling (CC) potentially improve load inequity and amplify resource consumption. Not at all capable to hold the job types since they does not

International Journal of Innovative Research in Science, Engineering and Technology*An ISO 3297: 2007 Certified Organization**Volume 7, Special Issue 8, September 2018***International Conference on Emerging Trends in Science, Technology and Mathematics (ICETSTM-2018)"****22nd August 2018****Organized by****Department of Computer Science, Parvathy's Arts and Science College, Dindigul, Tamilnadu, India**

rely on global synchronization. CloudSim (cloudbus.org) has been used to create the simulation environment. This paper adopts CloudSim1.0b to simulate the algorithm of task scheduling given above. CloudSim 1.0b provides a series of core function for the business and simulation of varied distributed computing ambience, mainly proper for simulation and research of task scheduling on cloud. To handle more complicated scenario involving dynamic factors such as dynamically changing cloud environment and other QoS attributes.

III. PROBLEM DESCRIPTION

In max-min interrupt was done by the newly arrived jobs. The Genetic algorithm creates latency and takes more waiting time for execute the job. Adaptive workflow strategy was not optimal for all kinds of jobs. Workflows with different execution time are entered into the cloud environment. In default the scheduler follow the First come first serve basis, based on their arrival time of the job. In Shortest job first they give more preference for small execution workflow. Round Robin algorithm the jobs are executed on the basis of random fashion. Benders Decomposition has more number of steps. It divides the module in to more sub modules it requires more time. It affects the quality of service the client wait more time for access the system. Needs and wants of the clients are increasing and varying one. It was uncertain one for client demands. In the existing system K-Means neighbor algorithm was used for predict the demand. Heuristics based algorithm did not give guarantee for effective utilization. Stochastic program are used for predict the uncertainty of client demands, it was difficult to implement.

IV. PROPOSED HIERARCHICAL JOB SCHEDULING ALGORITHM FOR CLOUD

The user submitted jobs are entered in to the cloud environment by the Cloud service provider. The proposed hierarchical scheduler algorithm classifies the jobs in different hierarchy on the basis of processor needed. The scheduler sorting arrived jobs in order. The jobs are further classified into three different levels namely top level, medium level and bottom level. If the processor is available they give priority for the top level jobs then medium and lower level jobs. If top level does not come they prefer medium level job. After the allocation of the entered jobs, they are executed in random fashion. While at the time of execution of jobs, normally the jobs attain the incomplete stage, complete stage and aborted stage based on the burst time of the job decided by the system. Finally the in-completed jobs are entered into the queue again after their finish their execution. From the figure1 shows the Architecture of Efficient Hierarchical job Scheduling Algorithm for Cloud Computing.

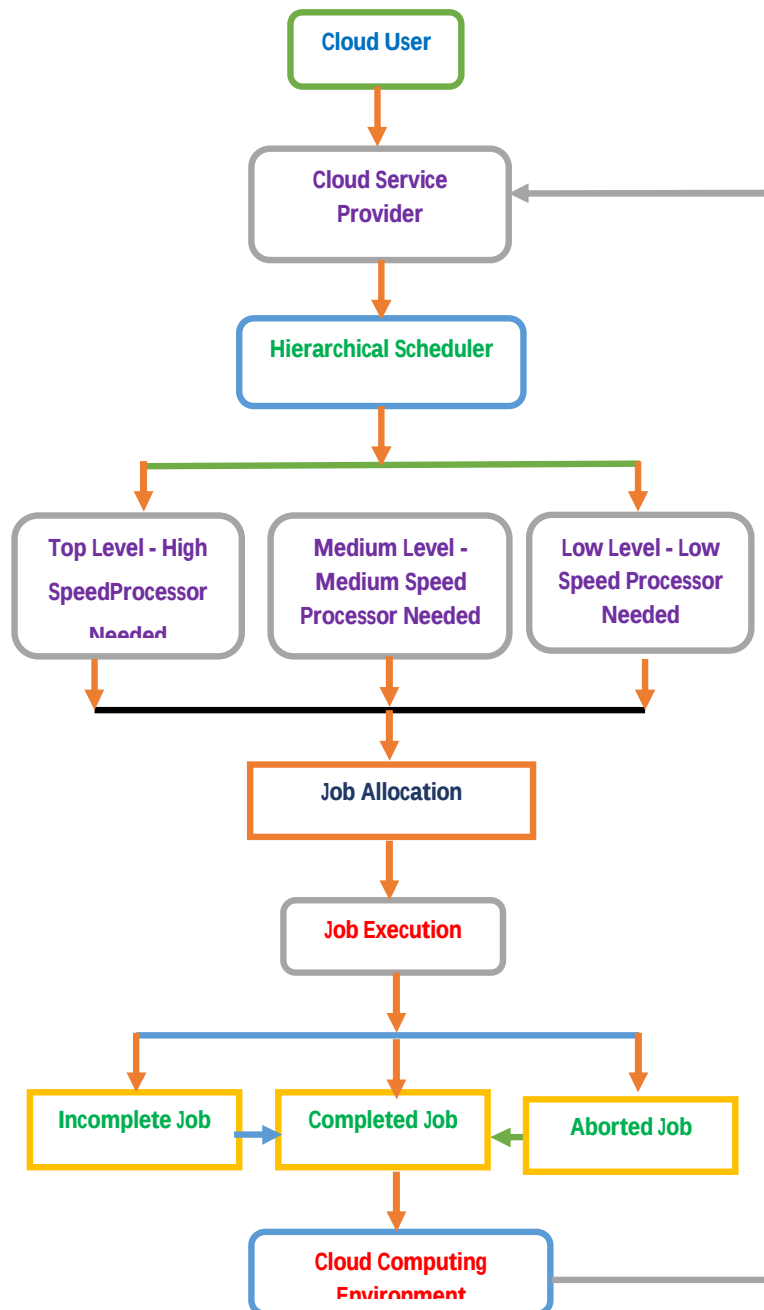


Figure1: Architecture of Efficient Hierarchical job Scheduling Algorithm for Cloud Computing

Algorithm for Hierarchical job Scheduling

1. User requested cloudlets are entered into the cloud.
2. Sorted the cloudlet based on their descending order of their burst time.
3. Mapping the server with highest burst time that allocate to the top level that needs high speed processor.
4. Medium burst time are allocated in medium speed processor.
5. Low burst time are allocated to low speed processor.

6. Based on their free availability of the cloud server the remaining cloudlets are executed.

V. RESULTS AND FUTURE ENHANCEMENT

Based on the execution of available data set using Cloud Sim 3.0, the proposed algorithm gives better results. The turn-around time and load balance of processor are used in optimum manner. The future work will be based on stochastic and artificial intelligence based scheduling for Cloud Environment.

VI. CONCLUSION

Efficient Resource Allocation and Load Balance Management is the foremost objective of this proposed work. Scheduling aims at full the work that maximizes cloud server resources in a balanced way. The proposed scheduling algorithm allocates the workload in a diversified manner from variety of cloudlet. Finally hierarchical job scheduling algorithm for Cloud Computing uses the system resources in an efficient method.

REFERENCES

- [1] AV.Karthick, E.Ramaraj, R.Kannan. (2013). An efficient Tri Queue job Scheduling using dynamic quantum time for cloud environment. IEEE ICGCE.
- [2] AV.Karthick, E.Ramaraj, R.Ganapathy Subramanian. (2014). An Efficient Multi Queue Job Scheduling for Cloud Computing. IEEE Computer Society-WCCCT (pp. 164-166).
- [3] AV.Karthick, K.Muthu Pandi. (2016). An overview of cost provisioning strategies for Cloud Computing. International Journal of Advanced Research in IT and Engineering. Vol. 5. No. 3. (pp. 1 – 8).
- [4] A. Sodan. (2010). Predictive Space and Time-Resource Allocation in Parallel Job Scheduling for Clusters, Grids, Clouds. 39th International Conference on Parallel Processing Workshops (pp.313 – 322).
- [5] D.Daniel, S.P.Jeno Lovesum. (2011). A Novel Approach for Scheduling Service Request in Cloud with Trust Monitor. International Conference on Signal Processing, Communication, Computing and Networking Technologies (pp. 509 - 513).
- [6] Dan Tsafir, Dror G. Feitelson. (2006). The Dynamics of Backfilling: Solving the Mystery of Why Increased Inaccuracy May Help. IEEE International Symposium on Workload Characterization (pp.131 – 141).
- [7] Dan Tsafir, Dror G. Feitelson. (2007). Backfilling Using System-Generated Predictions Rather than User Runtime Estimates. IEEE Transactions on Parallel and Distributed Systems (pp. 789 - 803).
- [8] Dejan Perkovic Peter J. Keleher. (2010). Randomization, Speculation, and Adaptation in Batch Schedulers. IEEE (pp.7 - 7).
- [9] Eitan Frachtenberg, Fabrizio Petrini. (2005). Adaptive Parallel Job Scheduling with Flexible Coscheduling. IEEE Transactions on Parallel and Distributed Systems (pp.1066 – 1077).
- [10] J. Octavio Gutierrez-Garcia and Kwang Mong Sim. (2011). A Family of Heuristics for Agent-based Cloud Bag-of-Tasks Scheduling. International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (pp.416 - 423).
- [11] Loganayagi B, S.Sujatha. (2010). Creating Virtual Platform for Cloud Computing. IEEE (pp.1 - 4).
- [12] Luqun Li (2009) An Optimistic Differentiated Service Job Scheduling System for Cloud Computing Service Users and Providers. Third International Conference on Multimedia and Ubiquitous Engineering (pp. 295 - 299).
- [13] Mohammad Sajid, Zahid Raza. (2015). Turnaround Time Minimization Based Static Scheduling Model Using Task Duplication for Fine-Grained Parallel Applications onto Hybrid Cloud Environment. IETE Journal of Research (pp.1 -13).
- [14] Rajkumar Kettimuthu, Vijay Subramani, Srividya Srinivasan Thiagaraja Gopal-asamy. (2002). Selective Preemption Strategies for Parallel Job Scheduling. Proceedings of the International Conference on Parallel Processing. IEEE (pp.602 - 610).
- [15] S.Selvarani, G.Sudha Sadhasivam. (2010). Improved Cost-Based algorithm for Task scheduling in Cloud Computing. IEEE (pp.1 - 5).
- [16] S.V.Sudha, K.Thanushkodi. An Approach for Parallel Job Scheduling using Nimble Algorithm (pp.1 - 9).
- [17] Sofia K. Dimitriadou. (2010). Job Scheduling in a Distributed System Using Backfilling with Inaccurate Runtime Computations. International Conference on Complex, Intelligent and Software Intensive Systems (pp. 329 - 336).
- [18] Srividya Srinivasan, Rajkumar Kettimuthu, Vijay Subramani, P. Sadayappan. (2002). Characterization of Backfilling Strategies for Parallel Job Scheduling. Proceedings of the International Conference on Parallel Processing Workshops (ICPPW'02).IEEE (pp. 514 - 519).
- [19] Sudha Sadhasivam, R. Jayarani.(2009). Design and Implementation of an efficient Two-level Scheduler for Cloud Computing Environment. International Conference on Advances in Recent Technologies in Communication and Computing (pp.585 - 586).
- [20] Su-Hui Chiang and Sangsue Vasupo-ngayya. (2008). Design and Potential Performance of Goal-Oriented Job Scheduling Policies for Parallel Computer Workloads. IEEE Transactions on Parallel and Distributed Systems (pp.1642 - 1656).
- [21] Suresh.A and Vijayakarthick.P. (2011). Improving scheduling of backfill algorithms using balanced spiral method for cloud metascheduler. IEEE-International Conference on Recent Trends in Information Technology. ICRTIT (pp. 624 - 627).
- [23] Yanyong Zhang, Jose A. Moreira. (2001). Impact of Workload and System Parameters on Next Generation Cluster Scheduling Mechanisms. IEEE Transactions on Parallel and Distributed Systems (pp.967 - 985).
- [24] Zhongni Zheng ,Rui Wang, Hai Zhong, Xuejie Zhang. (2011). An Approach for Cloud Resource Scheduling Based on Parallel Genetic Algorithm. IEEE (pp. 444 - 447). www.cloudbus.org/cloudsim.