Middle-East Journal of Scientific Research
24 (Special Issue on Innovations in Information, Embedded and Communication Systems): 47-50, 2016
ISSN 1990-9233; © IDOSI Publications, 2016
DOI: 10.5829/idosi.mejsr.2016.24.IIECS.23138

A Survey on Hybrid Task Scheduling in Cloud Environment

J. Caroline and R. Venkatesan

School of Computer Science and Technology, Karunya University, Coimbatore, India

Abstract: Cloud Computing is one of the prominent technologies which expand the boundary of the internet by using the centralized servers to maintain data and resources. In a cloud, task scheduling is the assignment of start and end time of the different tasks. Task scheduling helps to maximize the resource utilization and minimizing the execution time by distributing the jobs on a different virtual machine. The main objective of a Hybrid algorithm is to minimize the makespan, waiting time and reduce the task completion time. The proposed algorithm is hybrid scheduling algorithm to minimize the total execution time and the total cost to execute, also try to reduce the task completion time, makespan and waiting at the environment level. Most of the algorithms are concentrating any one task type to achieve improvement above mentioned parameters. In this paper various task scheduling strategies and their merits and demerits are discussed in detail.

Key words: Cloud Computing · Workflow Scheduling · Hybrid Task Scheduling

INTRODUCTION

Cloud Computing is a developing technology which maximizes the boundary of the internet to maintain data and resources by using high-performance servers. It is an internet based computing, whereby computers, the other shared resources, is provided to the other resources based on demand and pay-per- usage basis. Task scheduling is one of the major issues in cloud computing. It is an essential process to improve the overall efficiency in the cloud. Various algorithms are proposed by authors to improve the scheduling in the cloud. Examples are FIFO, Genetic algorithm, round robin algorithm etc. In order to achieve the optimal solution in task execution, any one of the scheduling algorithm can be adopted. Task scheduling is the management process to assign start and end time of different tasks. In a cloud, we have bounded with certain constraints for high performance. These constraints are classified into resource constraint and time constraint. The time constraint is referred to the limitations on the start and end times of each task in projects critical path. Resource constraint is referred to the limitations of equipment that is necessary to complete the project. Task scheduling is the part of cloud computing. It has two types: static scheduling and dynamic scheduling. Many algorithms are proposed for scheduling mechanism. In cloud computing, there is a queue of different tasks each task of different priority. Scheduler checks the priority of each job and allocates the job to different virtual machines according to their priority. The executing time is reduced, energy consumption is lowered and the resource utilization of the resources is improved by using Scheduling algorithm. Scheduling millions of task is a challenge to cloud computing environment. To minimize the total cost, optimal number of systems in the cloud should be decided by the resource allocation.

Cloud computing allows the user to dynamically specify the requirements of virtual machine. In order to efficiently and cost effectively schedule the tasks and data of applications on to these cloud computing environments, application scheduler has different policies that vary according to the objective function such as minimized execution time, low cost, balance the load on resources.

Literature Review: A literature survey has been conducted to study the different methods used for different scheduling in cloud computing. The reference paper is studied and compared with their advantages and disadvantages. McGough *et al*. [1] have proposed a number of policy decisions which can be applied to a virtual cluster to reduce the overall cost and the effect of these policies have on the users of the cluster. Lakra *et al*. [2] have proposed a multi-objective task scheduling algorithm for mapping tasks to a VMs in order to improve

Corresponding Author: J. Caroline, School of Computer Science and Technology, Karunya University, Coimbatore, India.

Table 1: Comparison of Various Task Scheduling Techniques

Sl. No	Title	Methodology	Metrics	Challenges
1	Deadline-Constraint workflow scheduling algorithm for infrastructure as a service clouds [3]	PCP algorithm.	Cost optimization and execution time optimization. Normalized cost of IC-PCP is reduced from 9% to 1.9% Normalized cost of IC-PCPD2 is reduced from 9% to 2% The Normalized cost of IC-Loss is reduced from 14% to 2%.	To minimize the overall cost of workflows, while meeting each workflow's deadline.
2	Comparison of a cost-effective virtual cloud cluster with an existing campus cluster [1]	Simulation mechanism for QoS task execution.	Cost. The Overall cost of using the cloud (21.3%) is reduced to (30.1%).	The reduction in cost difference coupled with the vast reduction in overheads.
3	Multi-Objective task scheduling algorithm for cloud computing throughput optimization [2]	Multi-objective task scheduling algorithm	It provides the minimum overall execution time and increased throughput.	Cloud computing works in real time but for task scheduling single criteria based algorithm may not be the suitable one.
4	An efficient approach to genetic algorithm for task scheduling in cloud computing environment [4]	Meta-heuristic based scheduling algorithm	Minimizes makespan. Modified Genetic Algorithm is better than Standard Genetic Algorithm 600(SGA)>550(MGA).	This algorithm is not supporting runtime scheduling and also not considering the user's quality of service.
5	Cost-aware challenges for workflow scheduling approaches in cloud computing environments [5]	Cost-aware Work Flow Scheduling approach	Improves scalability. Budget 15% makespan 31%	Less attention on availability and security aspects than others such as makespan.
6	Multi-Objective energy-efficient workflow scheduling using list-based heuristics [6]	Pareto-based multi-objective workflow scheduling algorithm	Maximizes makespan and reduces energy consumption. The percentage of energy savings is only up to 7%.	To extend this algorithm to consider multi-tenant commercial cloud systems.
7	Fast workflow scheduling for grid computing based on a multi-objective genetic algorithm [7]	Multi-objective Genetic Algorithm (GA)	To optimize both workflow execution time and cost. Execution time is 0.7 and the cost value is 0.97 and in LNSGA, execution time is 1 and the cost value is 0.86.	To improve grid computing performance.
8	Resource-aware hybrid scheduling algorithm in heterogeneous distributed computing[8]	Resource-aware hybrid scheduling algorithm	Execution time has reduced from 400 for 3 jobs to 50 for 9 jobs.	To compare the tasks with no dependencies in the clustering phase, a metric was proposed.
9	An interoperability model for ultra large scale systems [9]	A maturity model for the interoperability of ultra large scale systems	Maximize scalability. Interoperability maturity model has increased from 3% to 4%.	The total execution time.
10	Multi-criteria and satisfaction oriented scheduling for hybrid distributed computing infrastructures [10]	Prometheus Scheduler algorithm.	To maximize expected trust and completion time. 25% threshold, which is irrelevant.	Cost
11	Cloud task scheduling based on ant colony optimization [11]	Ant Colony Optimization (ACO) algorithm.	Minimizing makespan. ACO outperforms FCFS and RR Algorithms. 1010(FCFS)>1000(RR)>700(ACO)	The effect of precedence between tasks and load balancing will not be considered.
12	Deadline guarantee enhanced scheduling of scientific workflow applications in grid [12]	A novel probability evaluation based scheduling algorithm.	Maximizes the scalability. Scheduling Algorithm for Deadline Sensitive Scientific Workflow (SADSSW) is better than VgrADS by 45.8% and performance fluctuation aware workflow scheduling algorithm(PFAS) by 36.9%	Focuses on the adaptability and predictability under deadline constraint.

Table 1: Continued

13	An approach to improve task scheduling in a decentralized cloud computing environment [13]	Multiple QoS Constrained Scheduling	Improves the resource utilization	Multi workflows to be addressed.
14	Interoperability evaluation models: A systematic review [14]	Interoperability evaluation models	To avoid consistency and redundancy.	QoS requirements
15	A trust service-oriented scheduling model for workflow applications in cloud computing [15]	Trust service-oriented workflow scheduling model.	Execution time, cost and trust. Execution time is reduced from 1.05 to 1.0 Execution cost is reduced from 1400 to 400	Uncertainty and unreliable environments

the throughput of the data center and reduce the cost without violating the SLA (Service Level Agreement). Alkhanak *et al*. [5] have proposed to facilitate researchers in selecting appropriate cost-aware WFS approaches from the available pool of alternatives. In QoS challenges, researchers have paid less attention to availability and security aspects than others such as makespan. Durillo *et al*. [6] have proposed a new Pareto-based multi-objective workflow scheduling algorithm a stretching to an existing state-of-the-art heuristic capable of computing a set of trade-off optimal solutions in terms of makespan and energy efficiency. Rius *et al*. [7] have proposed a multi-objective Genetic Algorithm (GA) is proposed to improve grid computing performance. A new multi-objective genetic algorithm-based technique was proposed to optimize both workflow execution time and cost. Vasile *et al*. [8] have proposed hierarchical clustering of the available resources into groups in the allocation phase. The approach considers clustering of the available resources and received tasks, before the process of resource allocation. Rezaei *et al*. [12] have proposed with Multiple QoS Constrained Scheduling Strategy of Multi-Workflows to address this problem. Efficient task scheduling system can meet user requirements and thereby enhance the overall performance of the cloud. Choudhary *et al.* [14] has proposed the existing interoperability evaluation models and performs the analysis among their findings. The structuring interoperability evaluation model is ensuring consistency and avoiding redundancy.

Table 1 Shows comparison of various task scheduling techniques.

Proposed System: In the proposed system, the system presents an Optimization algorithms based on the behavior of birds (PSO) and honeybee are well known and widely used. PSO is known to be suitable for function optimization problems where the goal is to search for the optimum value of a function. Scheduling refers to the set of policies to control the order of work to be performed by a computer system. The performance of cloud computing is mainly affected by Resource allocation and scheduling of resources.

Honeybee Algorithm: The food foraging behavior of swarms of honey bees is mimiced by the Bees Algorithm which is a population-based algorithm. The Bees algorithm starts with the scout bees being randomly placed in the search space. The bees that have the highest fitness are chosen as selected bees and sites visited by them are chosen for static neighborhood sampling by evaluating the fitness of the scout bees.

Particle Swarm Optimization: Particle Swarm Optimization (PSO) is an arithmetical method that gives solutions while optimizes a problem by trying to improve a candidate solution. PSO evaluates a problem by having a number of candidate solutions, which is known as particles and these particles moving around in the search-space fitting to the particle's position and velocity. The movement of each particle is affected by its local-best known position and is also guided as to the best-known positions in the search-space, these are updated as better positions and then they are found by other particles. This is expected to move the best particle termed as swarm toward the best solutions. Particle Swarm Optimization algorithm has lots of features like good convergence rate, less expensive, easy to apply in a different scenario and simple to implement.

Equations: Makespan=$CTmax_{i,j}$ | $i \in T_i$=1, 2, …n and $j \in VM_j$=1, 2, …m Where $CTmax_{i,j}$ is the maximum which can be defined as the time for completing cloudlet i on a virtual machine j. Let VM= VM1, VM2, ….VMm is the number of m virtual machines that must be processed n tasks represented by the group T=T1, T2, … Tn. Each virtual machines are parallel and independent, the schedule independent tasks to these VMs, the Processing of that task on a virtual machine cannot interrupt, we denote end time of a task Ti by CTij. Our aim is to reduce the Makespan which can be denoted as CTmax, the runtime of each task for each virtual machine must be calculated for the purpose of scheduling, if the processing speed of virtual machine VMj is PSj, then the processing time for cloudlet Pi can be calculated by an equation.

$$P_{ij}=C_iPS_j \quad (1)$$

where Pij the processing is a time of task Pi by virtual machine VMj and C_i is the computational complexity of the task P_i. The processing time for each task in the virtual machine can be calculated by an equation.

$$Pj= Pij=1, \ldots, m \quad (2)$$

The processing time of a task varies from one VM to another based on the virtual machines speed. The processing time of a job varies from one VM to another based on the VM speed.

CONCLUSION

In this paper, we have described various task scheduling strategies and their challenges. The approach considers mapping of the search module and task scheduling using mapping function, before the phase of optimization module and finally the allocation module will be done. The techniques should provide minimum makespan to cloud providers and in the same way to reduce cost, time in the cloud providers perspective.

REFERENCES

1. McGough, A.S., M. Forshaw, C. Gerrard, S. Wheater, B. Allen and P. Robinson, 2014. Comparison of a cost-effective virtual cloud cluster with an existing campus cluster, Future Generation Computer Systems, 41: 65-78.
2. Lakra, A.V. and D.K. Yadav, 2015. Multi-Objective Tasks Scheduling Algorithm for cloud computing Throughput Optimization, (ICCC-2015) Procedia Computer Science, 48: 107-113.
3. Abrishami, Saeid Naghibzadeh, M.E.D.H., 2013. Deadline-constrained workflow scheduling algorithms for infrastructure as a service clouds, Future Gener. Comput. Syst., 29(1): 158-169.
4. Kaur, S. and A. Verma, 2012. An efficient approach to a genetic algorithm for task scheduling in cloud computing environment, Internat. J. Inform. Technol. Comput. Sci., 4(10): 74-79.
5. Alkhanak, E.N., S.P. Lee and S.R. Khan, 2015. Cost-aware challenges for workflow scheduling approaches in cloud computing environments: Taxonomy and opportunities, Future Generation Computer Systems, 50: 3-21.
6. Durillo, J.J., V. Nae and R. Prodan, 2014. Multi-objective energy-efficient workflow scheduling using list-based heuristics, Future Gener. Comput. Syst. (ISSN: 0167-739X), 36(0): 221-236.
7. Khajemohammadi, H., A. Fanian and T.A. Gulliver, 2013. Fast workflow scheduling for grid computing based on a multi-objective genetic algorithm, in: Communications, Computers and Signal Processing, PACRIM, 2013 IEEE Pacific Rim Conference on, IEEE, 2013, pp: 96-101.
8. Vasile, M.A., F. Pop, R.L. Tutueanu, V. Cristea and J. Kolodziej, 2015. Resource-aware hybrid scheduling algorithm in heterogeneous distributed computing, Future Generation Computer Systems, 51: 61-71.
9. Rezaei, R., T.K. Chiew and S.P. Lee, 2014. An interoperability model for ultra large scale systems, Adv. Eng. Softw., 67: 22-46.
10. Moca, M., C. Litan, G.C. Silaghi and G. Fedak, 2015. Multi-criteria and satisfaction oriented scheduling for hybrid distributed computing infrastructures. Future Generation Computer Systems.http://dx.doi.org/10.1016/j.future.2015.03.022.
11. Tawfeek, M., A. Sisi, A. Keshk and F. Torkey, 2015. Cloud task scheduling based on Ant Colony Optimization, The International Arab Journal of Information Technology, 12(2).
12. Yan, C., H. Luo, Z. Hu, X. Li and Y. Zhang, 2013. Deadline guarantee enhanced scheduling of scientific workflow applications in a grid, J. Comput., 8(4): 842 850.
13. Choudhary, V., S. Kacker, T. Choudhury and V. Vashisht, 2012. An approach to improve task scheduling in a decentralized cloud computing environment, Int. J. Comput. Technol. Appl., 3(1): 312-316.
14. Rezaei, R., T.K. Chiew, S.P. Lee and Z. Shams Aliee, 2014. Interoperability evaluation models: A systematic review, Comput. Ind., 65(1): 1-23.
15. Tan, W., Y. Sun, L.X. Li, G. Lu, T. Wang, 2014. A trust service-oriented scheduling model for workflow applications in cloud computing, IEEE Syst. J., 8(3): 868-878.