

SCHEDULING ALGORITHMS IN MULTI CLOUD ENVIRONMENTS – A SHORT REVIEW

A. Neela Madheswari*, M.Ramesh
CSE Department, Mahendra Engineering College, Namakkal, Tamilnadu, India
*Email: neelamadheswaria@mahendra.info

Received: 10th September 2022, Accepted: 3rd October 2022 and Published: 3rd October 2022

ABSTRACT

Background: Cloud computing is based on a pay-as-you-go model in which efficient allocation and management of resources plays a major role, since users are billed according to their actual usage; proper utilization of resources therefore remains an active research focus, and whether a single cloud or a multi-cloud environment is used, incoming jobs or tasks must be scheduled according to available resources and needs.

Objective: This short review examines the scheduling algorithms available for single- and multi-cloud environments and organizes them according to their categories.

Methods: The review surveys provider-specific schedulers (AWS Batch, Azure Scheduler, Kubernetes-based scheduling as used in Alibaba Cloud, and the Salesforce Apex scheduler) together with the general cloud scheduling framework, and classifies scheduling algorithms reported in the literature into task, resource, and workflow scheduling categories, including a comparison of eight representative task-scheduling schemes.

Results: Task scheduling algorithms (PRISM, throughput-optimal scheduling, hyper-heuristic scheduling, ACTS, SLPSO, EARH, agent-based scheduling, and EMO workflow scheduling) are compared in terms of advantages, drawbacks, and simulation platform; resource scheduling algorithms are grouped into dynamic, agent-based, and cost-optimization categories; and workflow scheduling algorithms are grouped into heuristic, metaheuristic, and hybrid categories.

Conclusion: Scheduling remains a central cloud-computing issue since it directly affects execution time, cost, and resource utilization; no single algorithm suits every requirement, and this review gives researchers a structured starting point for selecting scheduling strategies in single- or multi-cloud systems.

Keywords: Scheduling, task, Workflow, Resource, Multi-cloud

HIGHLIGHTS:

1. This short review on scheduling algorithms will help understand the fundamentals of cloud scheduling.
2. The review will help researchers getting started with Scheduling Algorithms in multi-cloud environments.
3. Eight task-scheduling schemes and the resource- and workflow-scheduling categories are classified and compared for quick reference.

INTRODUCTION

Cloud computing is one of the major research fields, utilized across industries to reduce IT costs. NIST defines cloud computing as “a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction” [1], with five key characteristics: on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service. Computing as a public utility was first envisioned by John McCarthy in 1961 and by Douglas Parkhill’s 1966 book *Challenges of the Computer Utility*; Amazon (2006), Microsoft Azure (2010), and IBM Smart Cloud (2011) later turned this vision into commercial services [2]. Cloud computing offers workflow features not found together in any single environment, letting users anywhere deploy applications on demand at a competitive, quality-of-service-based cost [3].

Cloud computing’s advantages – cost effectiveness, no software requirement, on-demand self-service, and easy online configuration and deployment – helped worldwide public cloud spending grow 18% in 2021 (partly pandemic-driven), from about US\$242,696 million in 2019 to a projected US\$362,263 million in 2022 [4]. In a multi-cloud system, each cloud has its own data center and scheduler assigning tasks to virtual machines using its own strategy – Amazon EC2 and Azure, for example, differ – and as multiple providers increasingly run applications together, a centralized management approach may be adopted to reach common goals [5]. Benefits of a multi-cloud environment include lower vendor lock-in, flexibility to distribute workloads, lower risk of DDOS attacks, and a resilient architecture from resources spread across clouds [6]. This paper reviews multi-cloud environments and the role of scheduling algorithms in achieving effective quality-of-service outcomes.

MATERIALS AND METHODS

Scheduling Algorithms Used by Cloud Service Providers

The AWS Batch scheduler evaluates when, where, and how to run jobs submitted to a job queue, running them in approximately submission order once dependencies are met [7]. Azure Scheduler runs jobs such as HTTP/S calls or Azure Storage queue messages on any schedule, ideal for recurring maintenance like log cleanup and backups, and can call services inside or outside Azure [8]. Kubernetes is now the de facto cloud-native platform for running workloads: in Alibaba Cloud, physical servers execute batch jobs while containers execute online services such as payment and transaction requests, with each offline workload comprising n jobs, tasks, and instances (Figure 1) [9]. Salesforce's Apex scheduler lets developers run Apex classes at a specified time for daily or weekly maintenance via Batch Apex, with up to 100 scheduled jobs at once [10].

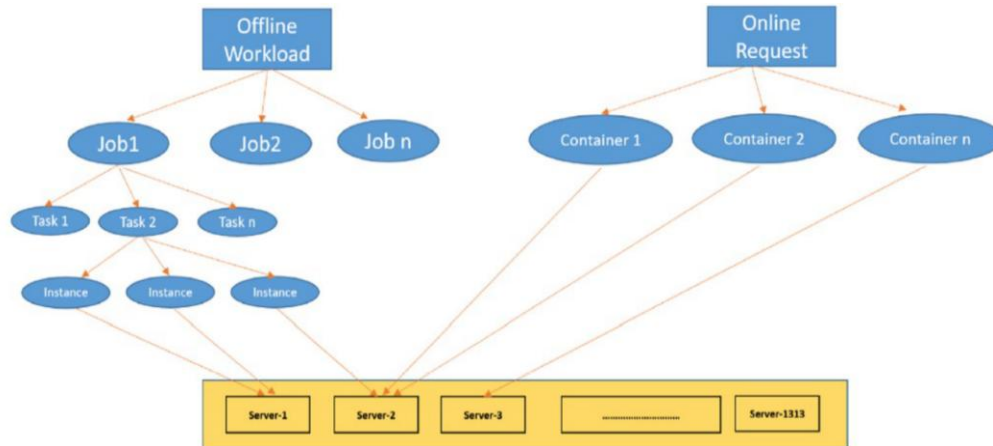


Figure 1: Offline workload and Online workload [9]

Cloud Scheduling Algorithms

Scheduling distributes resources among processes based on their execution demands, with the scheduler selecting which resource completes each job so that tasks run simultaneously across the cloud; poorly organized jobs reduce performance and prevent the cloud from delivering expected outcomes when processing large data volumes [11]. Scheduling methods fall into three groups: task scheduling (choosing suitable resources to execute a task, i.e. a user's query, within a required time), resource scheduling (distributing virtual resources among servers), and workflow scheduling (scheduling an entire job's workflow in an efficient order) [12].

RESULTS AND DISCUSSION

Task Scheduling Algorithms

There are a number of task scheduling algorithms proposed for cloud computing environments. A comparison of eight representative task scheduling algorithms, discussed in Mosleh et al. (2017) [13], is given in Table 1.

Table 1: Comparison of various task scheduling algorithms in cloud environments [13]

Scheduling Scheme	Advantages	Drawbacks	Simulation
PRISM (Phase and Resource Information-aware Scheduler for MapReduce)	Reduced overall job execution time	Deadlines not specified	Hadoop 0.20.2
Throughput-optimal Scheduling and Load-balancing Algorithm	Exact max-weight schedule chosen automatically at refresh times	Queue-length weighting relies on an assumption	Unknown
Hyper-heuristic scheduling algorithm	Reduces makespan and improves scheduling performance	High connection overhead	Hadoop
Adaptive cost-based task scheduling (ACTS)	Effective data access/fetch from data centers for each task	Does not schedule tasks with undetermined demands	CloudSim
Self-adaptive Learning PSO (SLPSO)-based scheduling	Better scheduling while maximizing IaaS profit and guaranteeing QoS	Lacks priority for deadline-constraint tasks, causing failures	Matlab 7.0
Energy-aware rolling-horizon (EARH) scheduling	Achieved real-time deadlines and reduced energy	Lacks updating of VM cycle count	CloudSim
Agent-based scheduling in virtualized clouds	Improved bidding process, reducing delay	Ignores communication/dispatch time, reducing performance	CloudSim
Evolutionary Multi-Objective (EMO) workflow scheduling	Reduced workflow scheduling	Ignores monetary cost and time overhead	Unknown

The scheduling model parameters considered across task scheduling algorithms include execution time, response time, makespan, throughput, resource utilization, load balancing, fault tolerance, energy consumption, scalability, performance, and quality of service [14].

Resource Scheduling Algorithms

Resource allocation is used to assign resources in an optimal way, and resource allocation algorithms are classified into three main categories: i) dynamic scheduling algorithms – five approaches including Linear Scheduling for

Tasks and Resources, round-robin-based resource selection, and Multiple Criteria Decision Analysis; ii) agent-based scheduling algorithms – seven approaches including negotiation-based, multi-agent-system, and market-based models; and iii) cost-optimization-based scheduling algorithms – three approaches including lightweight scaling and Compromised-Time-Cost scheduling [15].

Workflow Scheduling Algorithms

Workflow scheduling maps workflow tasks to VMs based on functional and non-functional requirements, since a workflow consists of interdependent tasks bound together by data or functional dependencies that must be considered during scheduling. Workflow scheduling algorithms are classified into three main categories: i) heuristic scheduling algorithms, spanning sixteen approaches such as Heterogeneous Earliest Finish Time, Priority Impact Scheduling, and various deadline-, cost-, QoS-, and fuzzy-clustering-based strategies; ii) metaheuristic scheduling algorithms, spanning six approaches such as Particle Swarm Optimization, Ant Colony Optimization, Simulated Annealing, and Genetic Algorithm; and iii) hybrid scheduling algorithms, spanning four approaches that combine these heuristic and metaheuristic techniques, such as Ant Colony Optimization with Max-Min scheduling and Genetic Algorithm with Best-fit and Round-robin [16].

CONCLUSION

Scheduling is one of the main issues in cloud computing: it plays a role in saving the time and price of executing tasks and in utilizing resources efficiently. This work has analyzed the different types of scheduling algorithms used in cloud environments, covering provider-specific schedulers as well as the task, resource, and workflow scheduling categories reported in the literature. This short review is helpful for researchers who are working towards better scheduling algorithms, whether a single cloud or a multi-cloud system is used; it is not possible to select a single scheduling algorithm for every requirement, since a number of considerations must be taken into account before choosing the scheduling algorithm best suited to a given user's needs.

REFERENCES

1. The NIST Definition of Cloud Computing. National Institute of Standards and Technology. <https://www.nist.gov/news-events/news/2011/10/final-version-nist-cloud-computing-definition-published>
2. Rajaraman, V. (2014). Cloud computing. *Resonance*, 19(3), 242–258.
3. Singh, J., & Gupta, D. (2017). A smarter multi queue job scheduling policy for cloud computing. *International Journal of Applied Engineering Research*, 12(9), 1929–1934.
4. Gartner. (2020). Gartner forecasts worldwide public cloud end-user spending to grow 18% in 2021. <https://www.gartner.com/en/newsroom/press-releases/2020-11-17-gartner-forecasts-worldwide-public-cloud-end-user-spending-to-grow-18-percent-in-2021>
5. Panda, S. K., Gupta, I., & Jana, P. K. (2015). Allocation-aware task scheduling for heterogeneous multi-cloud systems. *Procedia Computer Science*, 50, 176–184. <https://doi.org/10.1016/j.procs.2015.04.081>
6. Priyadarshini, A. (2020). Task scheduling algorithms in multi cloud environment. *International Journal of Recent Technology and Engineering*, 8(6). <http://www.doi.org/10.35940/ijrte.F7882.038620>
7. Amazon Web Services. Job scheduling. https://docs.aws.amazon.com/batch/latest/userguide/job_scheduling.html
8. Microsoft Azure. Azure Scheduler. <https://azure.microsoft.com/en-in/pricing/details/scheduler/>
9. Sharma, K. J., & Patel, N. M. (2021). Investigation of Alibaba Cloud dataset for resource management in cloud computing. In N. Chaubey et al. (Eds.), *COMS2 2021, CCIS 1416* (pp. 222–233). https://doi.org/10.1007/978-3-030-76776-1_15
10. Salesforce. Apex Scheduler. https://developer.salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/apex_scheduler.htm
11. Sana, M. U., & Li, Z. (2021). Efficiency aware scheduling techniques in cloud computing: A descriptive literature review. *PeerJ Computer Science*, 7, e509. <https://doi.org/10.7717/peerj-cs.509>
12. Malik, B. H., Amir, M., Mazhar, B., Ali, S., Jalil, R., & Khalid, J. (2018). Comparison of task scheduling algorithms in cloud environment. *International Journal of Advanced Computer Science and Applications*, 9(5). <https://dx.doi.org/10.14569/IJACSA.2018.090550>
13. Mosleh, M. A. S., Ghaleb, O. A. M., & Al-ariqi, H. D. E. (2017). A comparative study of various task scheduling algorithms in cloud environment. *Saba Journal of Information Technology and Networking*, 5(2). <http://ojs.sabauni.net/index.php/SJITN/article/view/87>
14. Ali, S. A., & Alam, M. (2016). A relative study of task scheduling algorithms in cloud computing environment. In 2016 2nd International Conference on Contemporary Computing and Informatics (IC3I) (pp. 105–111). <https://doi.org/10.1109/IC3I.2016.7917943>
15. Singh, A., & Malhotra, M. (2013). A comparative analysis of resource scheduling algorithms in cloud computing. *American Journal of Computer Science and Engineering Survey*. <https://www.primescholars.com/articles/a-comparative-analysis-of-resource-scheduling-algorithms-in-cloud-computing.pdf>
16. Masdari, M., ValiKardan, S., Shahi, Z., & Azar, S. I. (2016). Towards workflow scheduling in cloud computing: A comprehensive analysis. *Journal of Network and Computer Applications*, 66, 64–82. <http://dx.doi.org/10.1016/j.jnca.2016.01.018>