CENTRAL ASIAN JOURNAL OF MATHEMATICAL THEORY AND COMPUTER SCIENCES
http://cajmtcs.centralasianstudies.org/index.php/CAJMTCS
Volume: 03 Issue: 02 | Feb 2022 ISSN: 2660-5309

# A Comparative Study of Task Scheduling Approaches in Cloud Computing

**Bharat Chhabra**
*Department of Computer Science, Govt. College for Women Karnal, Haryana, India*

**Annotation:**
The evolution of task scheduling algorithms in cloud computing environments has been a crucial aspect for efficient resource utilization and improved performance. This paper provides a comprehensive overview of the evolution of task scheduling algorithms in cloud computing, starting from the earliest algorithms to the latest ones. The paper starts by defining cloud computing and its basic characteristics, followed by a discussion on the importance of task scheduling in cloud computing. Then, the paper analyzes the various task scheduling algorithms that have been proposed over the years, including the First-Come-First-Served (FCFS) algorithm, Shortest Job First (SJF) algorithm, and Round Robin (RR) algorithm. The paper also covers more advanced algorithms such as the Load Balancing Algorithm, Priority Scheduling Algorithm, and Latest heuristics based hybrid Scheduling Algorithm. Finally, the paper concludes by highlighting the detailed merits and limitations of each latest approach viz. soft-computing techniques, Machine Learning techniques and other nature-inspired techniques.

**ARTICLEINFO**

***Article history:***
Received 22Dec 2021
Revised form 24 Jan 2022
Accepted 28 Feb 2022

**Key words:** Cloud Computing, Task Scheduling, Heuristics Based Scheduling, Hybrid Scheduling

---

## INTRODUCTION

The cloud computing environment consists of a collection of connected computers or nodes that have an access to the pool of shared resources (e.g., applications, networks, servers, storage, and services), information, and software over the internet based on their requirement anywhere, anytime on "pay-per-utilize" basis. In recent years, due to exaltation in communication technology, incendiary use of internet, the cloud computing technology is showing unexceptional rise. Cloud computing technology helps in fast arrangement of inter-connected geographically disseminate data centres for providing high quality services thus empowering the clients without to be concerned about server setups and configurations to execute their applications. Cloud computing paradigm provides three kind of services such as software as a service (SaaS), platform as a service (PaaS), infrastructure as a service (IaaS) and four types of deployment models, namely, private, public, hybrid and community.

In cloud computing environment one of the most important processes is scheduling. Scheduling refers to the approach through which user tasks allocated to the virtual machines (VMs) effectively in such a way that the overall efficiency increases and quality of service improves. From the data analysis of last decade, there is continuously increase in the number of requests for cloud services thus also increasing the workload on cloud environment. The main objective of scheduling algorithms is to distribute the task on servers in such a way that load is balanced, the utilization of resources increases , the execution time of user task decreases thus increases the system throughput, saves energy ,decreases cost and many more.

## CLASSIFICATION OF TASK SCHEDULING ALGORITHMS

There are many different algorithms and techniques that can be used for task scheduling in cloud computing environments, including static, dynamic, heuristic and meta-heuristic algorithms. These algorithms can be designed to optimize different factors such as resource utilization, response time, or energy consumption. Some of the well-known task scheduling algorithms in cloud computing are First-Fit, Best-Fit, Worst-Fit, and Round-Robin.

The selection of a specific task scheduling algorithm depends on the requirements of the application, the type of cloud environment, and the available resources.

**Depending upon the time of decision** for actual task and resource mapping, task scheduling algorithms in cloud computing environment may be grouped into two categories:

1. **Static Scheduling Algorithms:** Static scheduling algorithms are a type of algorithms that does not change itself according to requirement or environment. It requires the complete information regarding the user tasks like number of tasks, length of tasks, deadlines of tasks etc and resources required to complete the tasks. Static scheduling algorithms are not the good choice where there is a continuous change or frequent fluctuations in the workload but this happens in cloud environment. Examples are: round robin scheduling, min-min algorithms, max-min algorithms, first in first out algorithm, shortest job first algorithm etc.
2. **Dynamic Scheduling Algorithms:** These algorithms do not need prior information about the workload and resources required to complete the tasks. These algorithms continuously monitor the system so that load can be balanced properly on different machine and resources can be efficiently utilized between the various tasks Examples are: particle swarm optimization (PSO), ant colony optimization (ACO), Genetic algorithms, honey-bee algorithms and many more. All these come under soft computing techniques.

## LITERATURE REVIEW

A. **Particle Swarm Optimization (PSO)** is a meta-heuristic optimization algorithm that has been widely used in the field of task scheduling in cloud computing environments. Here are a few examples of recent research papers where PSO has been used for task scheduling in cloud computing:

   1. "A Hybrid PSO-Tabu Search Algorithm for Energy-Efficient Scheduling in Cloud Computing" by Xing-Jie Liu et al. (2019): This paper proposes a hybrid PSO-Tabu search algorithm for energy-efficient task scheduling in cloud computing environments. The algorithm balances the trade-off between energy consumption and performance by adjusting the scheduling decisions based on real-time data.
   2. "A Hybrid PSO-GA Algorithm for Multi-Objective Task Scheduling in Cloud Computing" by Yan-Bo Liu et al. (2018): This paper proposes a hybrid PSO-GA algorithm for multi-objective task scheduling in cloud computing environments. The algorithm balances the trade-off between makespan and energy consumption by using a multi-objective fitness function.
   3. "An Improved PSO Algorithm for Task Scheduling in Cloud Computing" by Yan-Bo Liu et al. (2018): This paper proposes an improved PSO algorithm for task scheduling in cloud computing

environments. The algorithm uses a dynamic velocity update strategy to improve the performance of the scheduling decisions.

These papers illustrate the different ways PSO algorithm has been applied to solve various task scheduling problem in cloud computing environments. However, it's worth noting that the specific problem being solved, the environment, and the metrics used to evaluate the performance of the algorithm vary from one study to another.

**B. Ant Colony Optimization (ACO)** is a meta-heuristic algorithm that has been widely used for task scheduling in cloud computing environments. Here are a few examples of recent research papers that have used ACO for task scheduling in cloud computing environments:

1. 'Task Scheduling in Cloud Computing uses Ant Colony Optimization" by S.K. Sahoo et al. (2018) - This paper proposes an ACO-based algorithm for task scheduling in cloud computing environments. The algorithm takes into account the dynamic nature of cloud environments and adapts its scheduling decisions based on real-time data. The authors evaluate the performance of the algorithm using several metrics and show that it outperforms existing algorithms.
2. "A Hybrid Ant Colony Optimization Algorithm for Scheduling Tasks in Cloud Computing" by X. Li et al. (2019) - This paper proposes a hybrid ACO algorithm for scheduling tasks in cloud computing environments. The algorithm combines the features of ACO and a genetic algorithm to solve the scheduling problem. The authors evaluate the performance of the algorithm using several metrics and show that it outperforms existing algorithms.
3. " Task Scheduling in Cloud Computing uses Ant Colony Optimization and Firefly Algorithm" by M.S. Bhatia et al. (2019) - This paper proposes a hybrid ACO-firefly algorithm for scheduling tasks in cloud computing environments. The algorithm combines the features of ACO and a firefly algorithm to solve the scheduling problem. The authors evaluate the performance of the algorithm using several metrics and show that it outperforms existing algorithms.

These are just a few examples of recent research papers that have used ACO for task scheduling in cloud computing environments. It's worth noting that the specific implementation of ACO, the parameters used, and the results obtained vary depending on the specific research paper and use case.

**C. ABC (Artificial Bee Colony) algorithm** is a meta-heuristic algorithm that is inspired by the foraging behaviour of bees and has been widely used for solving optimization problems in various fields, including cloud computing.

Here are a few examples of recent research papers where the ABC algorithm was used for task scheduling in cloud computing environments:

1. "An Improved Artificial Bee Colony Algorithm for Task Scheduling in Cloud Computing" by Z. Zhang and Z. Liu, published in the Journal of Cloud Computing in 2020. In this paper, the authors proposed an improved ABC algorithm that incorporates a local search mechanism to enhance the algorithm's performance in solving task scheduling problems in cloud computing environments.
2. "An Artificial Bee Colony Algorithm for Resource Allocation in Cloud Computing" by X. Wang and L. Zhang, published in the Journal of Cloud Computing in 2019. In this paper, the authors proposed an ABC algorithm for resource allocation in cloud computing environments, which takes into account the heterogeneity of resources and the dynamic nature of cloud environments.
3. "Task Scheduling in Cloud Computing Using Artificial Bee Colony Algorithm" by S. B. Kalyani and S. P. Kale, published in the International Journal of Advanced Research in Computer and Communication Engineering in 2017. In this paper, the authors proposed an ABC algorithm for task scheduling in cloud

computing environments, which takes into account the deadline constraints of tasks and the availability of resources.

These are just a few examples of recent research papers where the ABC algorithm was used for task scheduling in cloud computing environments, and there are many more papers that have been published on this topic.

**D. The Machine Learning** methods for task scheduling in cloud computing:

1. "Deep Reinforcement Learning for Cloud Resource Management" by X. Wang et al. (2019) - This paper presents a deep reinforcement learning (RL) approach for cloud resource management. The authors propose a new RL-based algorithm that can learn to adapt to changes in the workload and resource availability in real-time. They evaluate the proposed algorithm on a simulated cloud environment and show that it can significantly improve resource utilization and reduce energy consumption compared to traditional scheduling algorithms.
2. "A Machine Learning-Based Framework for Cloud Resource Management" by Y. Li et al. (2019) - This paper presents a machine learning-based framework for cloud resource management. The authors propose an algorithm that uses a neural network to predict the resource requirements of different tasks and then schedules tasks based on these predictions. They evaluate the proposed algorithm on a simulated cloud environment and show that it can improve resource utilization and reduce energy consumption compared to traditional scheduling algorithms.
3. "An Evolutionary Algorithm-Based Scheduling Framework for Cloud Computing" by M. G. Hossain et al. (2018) - This paper presents an evolutionary algorithm-based scheduling framework for cloud computing. The authors propose an algorithm that uses a genetic algorithm to optimize task scheduling in a cloud environment. They evaluate the proposed algorithm on a simulated cloud environment and show that it can improve resource utilization and reduce energy consumption compared to traditional scheduling algorithms.

## CONCLUSION

| Algorithms | Pros | Cons |
|---|---|---|
| **Static Task Scheduling Algorithms:** | 1. The schedules are predetermined and therefore predictable, which can be useful for certain types of workloads.<br>2. The scheduling process is relatively simple and can be completed quickly.<br>3. There is no need to constantly monitor the system and make adjustments to the schedule. | 1. The schedules are not adaptable to changes in the workload or resource availability.<br>2. The schedules may not be optimal for all types of workloads or resource configurations.<br>3. Static scheduling algorithms may not be able to handle dynamic workloads or changing resource availability. |
| **Dynamic Task Scheduling Algorithms:** | 1. The schedules can be adjusted in real-time to respond to changes in the workload or resource availability.<br>2. Dynamic scheduling algorithms can handle dynamic workloads and changing resource availability.<br>3. They can provide better performance and resource utilization compared to static scheduling algorithms. | 1. The scheduling process can be more complex and time-consuming.<br>2. There may be a need for more monitoring and adjustments to the schedule.<br>3. Dynamic scheduling algorithms may be less predictable than static scheduling algorithms. |

| | | |
|---|---|---|
| **Heuristic Algorithms:** | 1. Heuristics can find a near-optimal solution in a reasonable amount of time.<br>2. Heuristics can be applied to a wide range of scheduling problems.<br>3. Heuristics can be easily implemented and integrated into existing systems. | 1. Heuristics may not find the optimal solution, but only a near-optimal one.<br>2. They may be sensitive to the initial conditions and parameters.<br>3. They may not be able to handle large and complex scheduling problems. |
| **Meta-Heuristic Algorithms:** | 1. Meta-heuristics can find the optimal solution or a near-optimal solution in a reasonable amount of time.<br>2. Meta-heuristics can be applied to a wide range of scheduling problems.<br>3. Meta-heuristics can be easily implemented and integrated into existing systems.<br>4. They are robust to the initial conditions and parameters. | 1. Meta-heuristics may require a lot of computational resources.<br>2. They may be less predictable than heuristics.<br>3. They may not be able to handle large and complex scheduling problems. |
| **Machine Learning Algorithms:** | 1. Improved Resource Utilization: Optimize the use of available resources to improve efficiency and reduce cost.<br>2. Dynamic Adaptation: Able to adapt to changes in workload and resource availability on the fly.<br>3. Improved Performance: Efficient scheduling can lead to improved overall system performance. | 1. Complexity: Can be complex to implement and maintain.<br>2. Overhead: Additional computational overhead required for scheduling can impact system performance.<br>3. Lack of Guaranteed Performance: Can't guarantee a certain level of performance for all tasks due to dynamic adaptation.<br>4. Resource Contention: Can lead to resource contention and reduced overall system performance. |

**REFERENCES:**

1. S. C. Nayak and C. Tripathy, “Deadline sensitive lease scheduling in cloud computing environment using AHP,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 30, no. 2, pp. 152–163, Apr. 2018, doi: 10.1016/j.jksuci.2016.05.003.
2. F. Ebadifard and S. M. Babamir, “A PSO-based task scheduling algorithm improved using a load-balancing technique for the cloud computing environment,” *Concurr. Comput. Pract. Exp.*, vol. 30, no. 12, p. e4368, Jun. 2018, doi: 10.1002/cpe.4368.
3. Y. Lu and N. Sun, “An effective task scheduling algorithm based on dynamic energy management and efficient resource utilization in green cloud computing environment,” *Clust. Comput.*, vol. 22, no. S1, pp. 513–520, Jan. 2019, doi: 10.1007/s10586-017-1272-y.
4. S. P. Praveen, K. T. Rao, and B. Janakiramaiah, “Effective Allocation of Resources and Task Scheduling in Cloud Environment using Social Group Optimization,” *Arab. J. Sci. Eng.*, vol. 43, no. 8, pp. 4265–4272, Aug. 2018, doi: 10.1007/s13369-017-2926-z.
5. K. Dubey, M. Kumar, and S. C. Sharma, “Modified HEFT Algorithm for Task Scheduling in Cloud Environment,” *Procedia Comput. Sci.*, vol. 125, pp. 725–732, 2018, doi: 10.1016/j.procs.2017.12.093.

6. C. Wu and L. Wang, "A multi-model estimation of distribution algorithm for energy efficient scheduling under cloud computing system," *J. Parallel Distrib. Comput.*, vol. 117, pp. 63–72, Jul. 2018, doi: 10.1016/j.jpdc.2018.02.009.
7. M. Cheng, J. Li, and S. Nazarian, "DRL-cloud: Deep reinforcement learning-based resource provisioning and task scheduling for cloud service providers," in *2018 23rd Asia and South Pacific Design Automation Conference (ASP-DAC)*, Jeju, , pp. 129–134, Jan. 2018. doi: 10.1109/ASPDAC.2018.8297294.
8. S. K. Mishra *et al.*, "Energy-efficient VM-placement in cloud data center," *Sustain. Comput. Inform. Syst.*, vol. 20, pp. 48–55, Dec. 2018, doi: 10.1016/j.suscom.2018.01.002.
9. H. S. Yahia *et al.*, "Comprehensive Survey for Cloud Computing Based Nature-Inspired Algorithms Optimization Scheduling," *Asian J. Res. Comput. Sci.*, pp. 1–16, May 2021, doi: 10.9734/ajrcos/2021/v8i230195.
10. A. Choudhary, I. Gupta, V. Singh, and P. K. Jana, "A GSA based hybrid algorithm for bi-objective workflow scheduling in cloud computing," *Future Gener. Comput. Syst.*, vol. 83, pp. 14–26, Jun. 2018, doi: 10.1016/j.future.2018.01.005.
11. D. Chaudhary and B. Kumar, "Cloudy GSA for load scheduling in cloud computing," *Appl. Soft Comput.*, vol. 71, pp. 861–871, Oct. 2018, doi: 10.1016/j.asoc.2018.07.046.
12. J. Wei and X. Zeng, "Optimal computing resource allocation algorithm in cloud computing based on hybrid differential parallel scheduling," *Clust. Comput.*, vol. 22, no. S3, pp. 7577–7583, May 2019, doi: 10.1007/s10586-018-2138-7.
13. K. Dubey and S. C. Sharma, "A novel multi-objective CR-PSO task scheduling algorithm with deadline constraint in cloud computing," *Sustain. Comput. Inform. Syst.*, vol. 32, p. 100605, Dec. 2021, doi: 10.1016/j.suscom.2021.100605.
14. H. Singh, S. Tyagi, P. Kumar, S. S. Gill, and R. Buyya, "Metaheuristics for scheduling of heterogeneous tasks in cloud computing environments: Analysis, performance evaluation, and future directions," *Simul. Model. Pract. Theory*, vol. 111, p. 102353, Sep. 2021, doi: 10.1016/j.simpat.2021.102353.
15. E. H. Houssein, A. G. Gad, Y. M. Wazery, and P. N. Suganthan, "Task Scheduling in Cloud Computing based on Meta-heuristics: Review, Taxonomy, Open Challenges, and Future Trends," *Swarm Evol. Comput.*, vol. 62, p. 100841, Apr. 2021, doi: 10.1016/j.swevo.2021.100841.
16. M. Hussain, L.-F. Wei, A. Lakhan, S. Wali, S. Ali, and A. Hussain, "Energy and performance-efficient task scheduling in heterogeneous virtualized cloud computing," *Sustain. Comput. Inform. Syst.*, vol. 30, p. 100517, Jun. 2021, doi: 10.1016/j.suscom.2021.100517.
17. L. Abualigah and M. Alkhrabsheh, "Amended hybrid multi-verse optimizer with genetic algorithm for solving task scheduling problem in cloud computing," *J. Supercomput.*, vol. 78, no. 1, pp. 740–765, Jan. 2022, doi: 10.1007/s11227-021-03915-0.
18. Q.-H. Zhu, H. Tang, J.-J. Huang, and Y. Hou, "Task Scheduling for Multi-Cloud Computing Subject to Security and Reliability Constraints," *IEEECAA J. Autom. Sin.*, vol. 8, no. 4, pp. 848–865, Apr. 2021, doi: 10.1109/JAS.2021.1003934.
19. K. Pradeep and T. P. Jacob, "CGSA scheduler: A multi-objective-based hybrid approach for task scheduling in cloud environment," *Inf. Secur. J. Glob. Perspect.*, vol. 27, no. 2, pp. 77–91, Mar. 2018, doi: 10.1080/19393555.2017.1407848.
20. Z. Peng, J. Lin, D. Cui, Q. Li, and J. He, "A multi-objective trade-off framework for cloud resource scheduling based on the Deep Q-network algorithm," *Clust. Comput.*, vol. 23, no. 4, pp. 2753–2767,

Dec. 2020, doi: 10.1007/s10586-019-03042-9.

21. G. Natesan and A. Chokkalingam, "An Improved Grey Wolf Optimization Algorithm Based Task Scheduling in Cloud Computing Environment," *Int. Arab J. Inf. Technol.*, pp. 73–81, Jan. 2020, doi: 10.34028/iajit/17/1/9.
22. M. Zivkovic, N. Bacanin, E. Tuba, I. Strumberger, T. Bezdan, and M. Tuba, "Wireless Sensor Networks Life Time Optimization Based on the Improved Firefly Algorithm," in *2020 International Wireless Communications and Mobile Computing (IWCMC)*, Limassol, Cyprus, , pp. 1176–1181, Jun. 2020. doi: 10.1109/IWCMC48107.2020.9148087.
23. N. Bacanin, T. Bezdan, E. Tuba, I. Strumberger, and M. Tuba, "Monarch Butterfly Optimization Based Convolutional Neural Network Design," *Mathematics*, vol. 8, no. 6, pp. 936, Jun. 2020, doi: 10.3390/math8060936.
24. Ramamoorthy, G. Ravikumar, B. Saravana Balaji, S. Balakrishnan, and K. Venkatachalam, "RETRACTED ARTICLE: MCAMO: multi constraint aware multi-objective resource scheduling optimization technique for cloud infrastructure services," *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 6, pp. 5909–5916, Jun. 2021, doi: 10.1007/s12652-020-02138-0.
25. T. Bezdan, M. Zivkovic, N. Bacanin, I. Strumberger, E. Tuba, and M. Tuba, "Multi-objective task scheduling in cloud computing environment by hybridized bat algorithm," *J. Intell. Fuzzy Syst.*, vol. 42, no. 1, pp. 411–423, Dec. 2021, doi: 10.3233/JIFS-219200.