Journal of Advanced Computer Applications
vol-01, issue-01, December, 2024

ISSN No.: XXXX-XXXX
Pages 74-83

JOURNAL OF ADVANCED COMPUTER APPLICATIONS

ISSN: XXXX-XXXX (Online)
Vol: 01 **Issue:** 01
Contents available at: https://www.swamivivekanandauniversity.ac.in/jaca/

A review on task scheduling algorithms in cloud computing towards energy efficiency

Sukriti Santra[1*]

[1]*Department of Computer Science and Engineering, Swami Vivekananda University, Barrackpore-700121, WB, INDIA*

ABSTRACT

The growing dependence on cloud computing, fueled by technological advancements, has led to a surge in energy consumption across cloud servers. Efficient resource scheduling within cloud data centers can reduce energy consumption. This paper provides a comprehensive overview of various task scheduling algorithms, encompassing traditional methods, heuristics, metaheuristics, and hybrid approaches. It delves into the challenges associated with workload distribution, resource allocation, and the dynamic nature of cloud environments. The significance of incorporating QoS parameters, such as deadline constraints and energy efficiency, is highlighted. Furthermore, the paper discusses the limitations and improvements of notable algorithms, such as Min-min and Max-min, and explores recent advancements in agent-based and heuristic scheduling methods.

Keyword: Cloud computing, cloud data centers, resource scheduling, green computing, energy efficiency.

* Authors for Correspondence: sukritis@svu.ac.in

I. INTRODUCTION

Cloud computing is a service-centric model providing on- demand access to computing resources, transforming IT pro- visioning and enabling global, distributed access from anydevice [1]. Cloud service providers are categorized into three layers: Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Platform as a Service (PaaS). IaaS offers virtualized resources like on-demand storage. PaaS providesa higher level of abstraction, making it easily programmable and allowing users to create applications without concerning themselves with specific hardware requirements. SaaS enablesusers to access software over the internet, freeing them from the responsibilities of software maintenance. In essence, IaaS provides virtualized storage, PaaS offers a programmable platform, and SaaS delivers software applications over the internet, relieving users of maintenance tasks [2]. Cloud computing serves as a versatile platform, providing access to resources, facilitating scalability, and enhancing operational efficiency for various application and services. Indeed, cateringto a growing demand for cloud services involves substantial energy consumption, contributing to carbon emissions. The environmental impact highlights the importance of adopting sustainable practices in cloud computing to mitigate adverse effects on the environment. To address environmental impact of cloud server energy consumption, various task scheduling algorithms have been proposed. These algorithms aim to optimize resource utilization, reduce energy consumption, andultimately contribute to a more sustainable and eco-friendly cloud computing environment. The rise in the digital economyis concomitant with data centers' increasing energy usage. Dueto the environmental impact of these high-energy-consuming entities, there is a focus on energy saving and emission reduction. As a result, improving cloud data centers' energy efficiency has become a major topic of research. Researchers are working hard to develop useful measurements and approaches for assessing energy efficiency in order to accomplishthis main objective [3].

In this comprehensive survey paper, we have endeavored toencompass task scheduling methodologies within the realm of cloud computing, with a specific focus on promoting energy efficiency.

The article is structured as follows: Section 2 entails a literature survey encompassing various task-scheduling algorithms. Section 3 delves into diverse task scheduling approaches, whileSection 4 offers insights from an energy efficiency perspective. Finally, Section 5 concludes the paper and outlines avenues forfuture research.

II. LITERATURE SURVEY

Several papers have been meticulously reviewed concerning task scheduling in cloud computing with a specific emphasis on enhancing energy efficiency. In a work Li Mao et. al [4]addressed the evolving challenges in cloud computing by introducing two algorithms, time-aware and energy-aware, designed for task scheduling in a heterogeneous environment. The proposed Energy-Performance Trade-Off Multi-Resource Cloud Task Scheduling Algorithm (ETMCTSA) combines these algorithms, allowing users to manage energy and per- formance based on their preferences, demonstrated through simulated experiments with favorable outcomes

compared to other scheduling methods.In another work Nirmal Kr. Biswas et. al [5] proposed a novel approach involving a New Linear Regression (NLR) prediction model, host load balancing, and Virtual Machine (VM) placement policies is proposed to mitigate energy consumption and Service Level Agreement violation in cloud data centers, demonstrating potential for building smart and sustainable environments for emerging Smart Cities. Author [6] et. al introduced the Energy-Efficient and Reliability-Aware Workflow Task Scheduling (EERS) algorithm for cloud environments, this approach optimizes energy conservation and system reliability throughfive sub-algorithms addressing task dependencies, communi- cation costs, sub-makespan definition, cluster-VM mapping, and slack reclamation. Evaluation on the WorkflowSim simu- lator with real-world scientific workloads demonstrates EERS outperforming existing approaches in energy efficiency and reliability optimization. In another work JK Jeevitha et. al [7] proposed Shortest Round Vibrant Queue (SRVQ) algorithm, acombination of Shortest Job First, Round Robin, and Vibrant Quantum, significantly reduces waiting times in the schedulingprocess and minimizes starvation. Through the collaborative use of DVFS and SRVQ, the research achieves a noteworthy 45% improvement in energy efficiency and a substantial 33% enhancement in Quality of Service (QoS) performance com- pared to existing algorithms. Author Sarita Simaiya et. al [8]in their paper introduces the ”EEPSA” method, a novel energy efficiency priority scheduling system for cloud computing that emphasizes pre-emptive scheduling and considers both optimalfit and system availability in routing requests to processing servers. The experimental analysis demonstrates the effec- tiveness of EEPSA in minimizing overall energy utilization compared to existing methods, evaluating parameters suchas energy usage, number of tasks migrated, and processing time. In another work H. Momeni et. al [9] introduce EaRTs, an energy-aware task scheduling approach for real- time applications in cloud computing. The proposed techniqueutilizes virtualization and consolidation to minimize energy consumption, enhance resource utilization, and meet task deadlines, featuring four algorithms that collectively demon- strate superior efficiency in terms of deadline hit ratio, resourceutilization, and energy consumption compared to other energy-aware real-time scheduling approaches. In a work Mohan Sharma et. al [10] proposed an energy-efficient independent task scheduler utilizing supervised neural networks, aiming to minimize makespan, energy consumption, execution overhead,and the number of active racks in cloud environments. Through extensive training with a genetic algorithm-generated dataset, our neural network achieves 99.9% accuracy, demonstrat- ing significant performance improvements over well-known approaches such as Genetic Algorithm, MinMIN-MINMin heuristic, and Linear regression-based energy-efficient taskschedulers in heavily and lightly loaded cloud environments.

III. DIFFERENT TASK SCHEDULING TECHNIQUES

Scheduling in cloud computing involves optimizing resource allocation for tasks based on quality of service parameters, utilizing either heuristic or meta-heuristic algorithms. Efficient resource provisioning algorithms are crucial to address challenges such as workload imbalance, over-provisioning,and under-provisioning, minimizing costs and optimizing re- source utilization [11]. Task scheduling in cloud computing environment can be static and dynamic. Task scheduling

algorithms can be classified as traditional scheduling, heuristic scheduling, meta-heuristic scheduling, and hybrid scheduling algorithms [12]. Classification is shown in Figure 1 below.

A) Traditional scheduling

Traditional task scheduling methods in cloud computing typically involve static or dynamic approaches. In static scheduling, tasks are assigned to resources at the beginning without considering runtime variations. Dynamic scheduling adapts to changing conditions during execution. Both methods aim to optimize resource utilization, minimize task completion time, and ensure adherence to quality of service (QoS) requirements. Classic algorithms like Round Robin, First Come First Serve (FCFS), and Priority Scheduling are examples of traditional methods used for task scheduling in cloud environments.

Heuristic scheduling

Heuristic task scheduling methods in cloud computing rely on rules of thumb and approximation techniques to make decisions quickly, especially in situations where finding an optimal solution is computationally complex. These heuristics aim to efficiently allocate tasks to virtual machines while considering factors like resource availability, load balancing, and minimizing task completion time.

Various heuristic techniques are employed in cloud computing for task scheduling, each tailored to specific optimization goals. Notable approaches include [11]:

Min-Min and Max-Min Algorithm

The Min-min scheduling algorithm minimizes expected completion time by iteratively assigning tasks to resources in two phases, calculating and selecting the task with the overall minimum expected completion time. However, it struggles to balance the load efficiently, often favoring small tasks during initial scheduling [13].

Contrastingly, the Max-min algorithm, commonly applied in distributed environments, prioritizes tasks with maximum expected completion time. It assigns these tasks to resources with the minimum overall execution time, repeating the pro- cess until all tasks are scheduled, and optimizing resource utilization.

In the Max-min algorithm, tasks are assigned based on a calculated matrix of expected completion time, selecting tasks with maximum expected completion time and minimum execution time for efficient resource allocation.

FCFS, SJF, and RR algorithm

In cloud computing, task scheduling is essential for optimizing resource utilization and performance. First Come First Serve (FCFS) prioritizes tasks based on arrival order, but it may lead to inefficiencies like the convoy effect. Shortest Job First (SJF) improves efficiency by

prioritizing shorter tasks, although accurate burst time prediction is crucial. Round Robin allocates fixed time slices to tasks, promoting fairness and preventing resource monopolization by longer tasks, but may incur higher turnaround times. The choice of scheduling algorithm depends on factors such as workload characteristics and system goals, with hybrid or adaptive approaches often used to address the dynamic nature of cloud environments.

Bin Packing

Bin packing scheduling in cloud computing involves efficiently assigning tasks to available resources to minimize wastage and optimize utilization. Analogous to packing items into bins, this approach aims to find an optimal arrangement of tasks across virtualized resources, such as virtual machines, to enhance efficiency and reduce operational costs. Various algorithms, including First Fit, Best Fit, and Worst Fit, are employed to allocate tasks to resources based on specific criteria. Efficient bin packing strategies are essential in cloud environments to ensure effective resource utilization, contributing to cost-effectiveness and improved overall system.

Deadline based scheduling

Deadline-based scheduling algorithms in cloud computing prioritize task execution based on specific time constraints or deadlines. The primary goals include meeting service level agreements (SLAs), minimizing the risk of missing deadlines, optimizing resource utilization, and enhancing overall system performance. These algorithms assign priorities to tasks based on their deadline require- ments and dynamically allocate resources to ensure timely completion. Factors like task urgency, resource availability, and system load are considered for effective scheduling deci- sions. Efficient deadline-based scheduling is crucial for time- sensitive applications, contributing to improved Quality of Ser- vice (QoS) and resource optimization in cloud environments. In [14]authors paper introduces DEESA, a Deadline-based Energy Efficient Scheduling Algorithm, designed for task and VM scheduling in cloud computing. Through dynamic queue classification of tasks and virtual machines, the proposed algorithm demonstrated in Cloudsim simulator effectively minimizes energy consumption and makespan time, surpassing the performance of existing scheduling algorithms.

QoS Based Scheduling

Quality of Service (QoS)-based scheduling is pivotal in addressing the escalating demand for efficient resource allocation in distributed and cloud computing environments. The focus is on enhancing task scheduling to ensure optimal utilization of available resources without overloading specific systems. Effective task scheduling not only improves cloud performance but also provides superior services to users. Various algorithms have been proposed for QoS-driven task scheduling, such as the QoS-guided Min- Min heuristic, which integrates adaptive scheduling with QoS guidance. Other approaches incorporate fixed-priority scheduling algorithms like Rate Monotonic and Deadline Monotonic, aiming to prioritize tasks based on their urgency. Additionally, QoS-aware algorithms categorize tasks based on attributes like user type, task type, size, and latency for more nuanced scheduling. These efforts emphasize the significance of QoS considerations in task scheduling to enhance resource utilization and meet diverse user requirements in cloud computing environments.

Agent and credit based scheduling algorithm

Agent Credit-Based Scheduling Algorithm is a proposed approach to task scheduling that leverages autonomous agents to enhance Quality of Service (QoS) parameters. A. Singh et al. introduced the Autonomous Agent-Based Load Balancing Algorithm (A2LB), where agents dynamically allocate resources to upcoming tasks, aiming to improve both response and execution time, along with scalability. Additionally, T. Thomas et al. put forth a credit-based algorithm as a solution to overcome limitations of the min-min algorithm. This algorithm calculates the average length of tasks and assigns credits based on the difference between the calculated average and individual task lengths. Tasks are then executed at resources according to their credit values, leading to improved make span time and resource utilization. The agent credit-based scheduling algorithm utilizes autonomous agents to make runtime decisions, contributing to the optimization of QoS parameters in cloud computing environments.

Meta-heuristic Based Scheduling

Metaheuristic algorithms have become widely popular in the last two decades due to their effectiveness in addressing complex and large computational problems. These algorithms possess valuable features such as problem independence, effi- cient exploration of search spaces for suboptimal solutions to NP-complete problems, and non-deterministic, approximation- based characteristics. Meta-heuristics, consisting of heuristics and randomization, are versatile tools applicable across vari- ous fields, showcasing high performance and adaptability to diverse problem domains. In the realm of cloud computing, where task scheduling often involves exploring a vast solution space to find optimal or near-optimal solutions, meta-heuristic algorithms play a crucial role. These algorithms, characterized by their non-deterministic and randomized nature, are partic- ularly suitable for NP-hard optimization problems. The use of meta-heuristic strategies in cloud environments allows for the efficient resolution of NP-complete problems, ensuring quick acquisition of sub-optimal solutions. [12]. Different meta- heuristic algorithms are mentioned below.

Particle Swarm Optimization Algorithm

Particle Swarm Optimization (PSO) has proven to be an effective metaheuristic algorithm for task scheduling in cloud computing. Utilizing the collective behavior of particles moving through a solution space, PSO aims to optimize objectives such as cost reduction, workflow efficiency, and resource utilization. Studies, such as Pandey et al.'s work on cost reduction and Juan et al.'s ap- proach to cloud storage optimization, showcase the algorithm's ability to refine task assignments for improved performance. Gomathi and Krishnasamy's hybrid PSO algorithm enhances resource utilization, while Alkayal et al. focus on multi- objective optimization, minimizing waiting time, and maximizing system throughput. Hybrid approaches, like the one proposed by Dordaie and Jafari Navimipour, demonstrate the adaptability of PSO in addressing various challenges in cloud task scheduling, highlighting its role in achieving efficient and effective solutions for complex cloud computing environments.

Ant Colony Optimization Algorithm

Ant Colony Optimization (ACO) has emerged as a powerful metaheuristic algorithm for optimizing task scheduling in cloud computing. Inspired by the foraging behavior of ants, ACO efficiently explores the solution space by using artificial ants to repre- sent potential task assignments. Pheromone trails guide the iterative construction of solutions, enabling the algorithm to converge towards optimal or near-optimal task schedules. ACOproves effective in minimizing make span, addressing dynamic changes in cloud environments, and accommodating multi-objective optimization goals such as execution time, cost, and resource utilization. Its decentralized and adaptive nature aligns well with the distributed and dynamic characteristics of cloud computing, making ACO a valuable tool for enhancing the efficiency of task scheduling in complex cloud environments.

Genetic Algorithm Based Task Scheduling Algorithm

Genetic Algorithms (GAs) have proven to be a robust optimization technique for task scheduling in cloud computing. Employing a chromosome representation for potential task schedules, GAs use genetic operations like crossover and mutation to iteratively evolve a population of schedules. The fitness of each schedule is evaluated based on defined objectives, such as minimizing makespan or optimizing re-source utilization. GAs demonstrate adaptability to dynamic environments, making them well-suited for cloud computing scenarios with varying workloads and resource availability. Studies have shown the efficacy of Genetic Algorithm-based task scheduling in achieving improved performance metrics, making it a valuable approach for optimizing task allocation in complex cloud environments.

IV. CONCLUSION AND FUTURE WORK

In conclusion, the optimization of task scheduling in cloud computing is a critical aspect that directly influences the overall efficiency and performance of cloud services. Var- ious scheduling algorithms, including traditional methods, heuristics, meta-heuristics, and hybrid approaches, have been explored to address the challenges of workload distribution, resource utilization, and meeting Quality of Service (QoS) requirements. While each algorithm has its strengths and limitations, ongoing research emphasizes the need for adaptive, energy-efficient, and scalable solutions to cater to the dynamic nature of cloud environments. Moreover, the incorporation of QoS parameters, such as deadline constraints and energy efficiency, has become increasingly crucial to ensure user satisfaction and environmental sustainability. Future work in this domain should focus on developing innovative algorithms that integrate machine learning techniques, consider security aspects, and address the evolving landscape of cloud computing technologies. Additionally, exploring the application of emerging technologies like edge computing and quantum computing in the context of task scheduling could open new avenues for improving performance and resource allocation in cloud environments. Overall, the field of cloud

task schedulingremains dynamic, presenting exciting opportunities for advancements that align with the evolving needs and challenges of cloud computing

REFERENCES

[1] A. Sunyaev and A. Sunyaev, "Cloud computing," *Internet Computing: Principles of Distributed Systems and Emerging Internet-Based Tech- nologies*, pp. 195–236, 2020.

[2] A. Amini Motlagh, A. Movaghar, and A. M. Rahmani, "Task scheduling mechanisms in cloud computing: A systematic review," *International Journal of Communication Systems*, vol. 33, no. 6, p. e4302, 2020.

[3] S. Long, Y. Li, J. Huang, Z. Li, and Y. Li, "A review of energy efficiency evaluation technologies in cloud data centers," *Energy and Buildings*, vol. 260, p. 111848, 2022.

[4] L. Mao, Y. Li, G. Peng, X. Xu, and W. Lin, "A multi-resource task scheduling algorithm for energy-performance trade-offs in green clouds," *Sustainable Computing: Informatics and Systems*, vol. 19, pp. 233–241, 2018.

[5] N. K. Biswas, S. Banerjee, U. Biswas, and U. Ghosh, "An approach towards development of new linear regression prediction model for reduced energy consumption and sla violation in the domain of green cloud computing," *Sustainable Energy Technologies and Assessments*, vol. 45, p. 101087, 2021.

[6] R. Medara and R. S. Singh, "Energy efficient and reliability aware workflow task scheduling in cloud environment," *Wireless Personal Communications*, vol. 119, no. 2, pp. 1301–1320, 2021.

[7] J. Jeevitha and G. Athisha, "A novel scheduling approach to improve the energy efficiency in cloud computing data centers," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, pp. 6639–6649, 2021.

[8] S. Simaiya, V. Gautam, U. K. Lilhore, A. Garg, P. Ghosh, N. K. Trivedi, and A. Anand, "Eepsa: Energy efficiency priority scheduling algorithm for cloud computing," in *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC)*. IEEE, 2021, pp. 1064– 1069.

[9] H. Momeni and N. Mabhoot, "An energy-aware real-time task schedul- ing approach in a cloud computing environment," *Journal of AI and Data Mining*, vol. 9, no. 2, pp. 213–226, 2021.

[10] M. Sharma and R. Garg, "An artificial neural network based approach for energy efficient task scheduling in cloud data centers," *Sustainable Computing: Informatics and Systems*, vol. 26, p. 100373, 2020.

[11] M. Kumar, S. C. Sharma, A. Goel, and S. P. Singh, "A comprehensive survey for scheduling techniques in cloud computing," *Journal of Network and Computer Applications*, vol. 143, pp. 1–33, 2019.

[12] 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 and Evolutionary Computation*, vol. 62, p. 100841, 2021.

[13] B. Kanani and B. Maniyar, "Review on max-min task scheduling algorithm for cloud computing," *Journal of emerging technologies and innovative research*, vol. 2, no. 3, pp. 781–784, 2015.

[14] S. K. Grewal and N. Mangla, "Deadline based energy efficient schedul- ing algorithm in cloud computing environment," in *2021 Fourth Inter- national Conference on*

Computational Intelligence and Communication Technologies (CCICT). IEEE, 2021, pp. 383–388.

[15] A. A. Khan and M. Zakarya, "Energy, performance and cost efficient cloud datacentres: A survey," *Computer Science Review*, vol. 40, p. 100390, 2021.