Journal of Global Research in Electronics and Communication
Volume 1, No. 8, August 2025
Available Online at: www.jgrec.info
RESEARCH PAPER

A Comparative Review of Heuristic vs Metaheuristic Techniques in Cloud Resource Optimization

Prof. (Dr.) Abid Hussain
Professor
School of Computer Application & Technology & Dean, Research
Career Point University
Kota
abid.hussain@cpur.edu.in, dean.research@cpur.edu.in

***Abstract*—The flexibility, scalability, and on-demand services offered by cloud computing have fundamentally altered the way computer resources are accessed and handled. Nevertheless, dynamicity and heterogeneity of cloud environment present a major problem in the optimization of resources efficiently. A key problem in cloud computing is optimizing cloud resources, which aims to improve performance, reduce operating expenses, and guarantee effective use of computational resources including CPU, memory, storage, network bandwidth, and energy. A thorough review of cloud resource categories and the main distinctions between scheduling and resource allocation procedures is provided in this work. It investigates the use of heuristic approaches that provide quick and easy solutions for resource management in predictable situations, such as Shortest Job First (SJF), Round-Robin (RR), Min-Min, Max-Min, and First-Come-First-Served (FCFS). Additionally, because they offer scalable and reliable optimizations in dynamic and diverse cloud environments, the study highlights metaheuristic algorithms including Simulated Annealing (SA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), and Genetic Algorithm (GA). A comparative analysis highlights the strengths and limitations of both strategies in terms of computational overhead, accuracy, scalability, and convergence speed. The paper concludes by identifying key future directions, including hybrid models, AI-driven techniques, energy-aware optimization, privacy-preserving strategies, and cloud-edge synergy, to address the evolving demands of next-generation cloud systems.**

Keywords*—Cloud Computing, Task Scheduling, Genetic Algorithm, Round-Robin (RR), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), Heuristic, Metaheuristic, Scalability

I. Introduction

The virtual machine and cloud server play important roles in cloud resource management. As a result of extensive execution time and resource failure, the utilization of resources is very high. Reducing energy usage and preventing resource failure are essential for swiftly managing cloud resources [1]. The remaining chores supported by the resource computer equipment plans for the next cloud upheld the delicate aspects of the outstanding loads. First, get the remaining tasks for the cloud planned so that the content, accessible resources, and cloud outstanding loads are mapped out as soon as possible to support the organizing techniques.

A common pool of reconfigurable computer resources, such as storage, servers, and applications, is made available on demand through cloud computing, which has become a paradigm shift in contemporary computing. Resource optimization is a crucial issue in cloud systems. It involves efficiently assigning and managing computing resources to provide scalability, cost-effectiveness, energy efficiency, and compliance with Quality of Service (QoS) and Service Level Agreements (SLAs) [2]. As cloud workloads become more dynamic, heterogeneous, and data-intensive, traditional static and rule-based optimization methods often fall short in delivering the required performance and adaptability.

The unpredictability of workloads in these situations makes optimizing cloud resources much more difficult. Because workload needs might vary greatly, it can be challenging to manually distribute resources in a way that strikes a balance between cost-effectiveness and performance [3][4]. Traditional resource management strategies, which rely on static rules or basic heuristics, cannot handle the complexity and dynamic nature of such workloads. Optimizing cloud environments to guarantee good performance while lowering costs requires the capacity to forecast future resource requirements and make real-time adjustments to allocations.

Ensure that computer resources are allocated, scheduled, and utilized as efficiently as possible. This is one of the primary concerns of cloud computing [5][6]. Resource optimization not only impacts the cost and performance of cloud services but also influences system responsiveness, energy consumption, and compliance with service-level agreements (SLAs). Effective resource management requires intelligent decision-making strategies that can handle dynamic workloads, heterogeneity, and multi-objective requirements.

Heuristic approaches have historically been employed to address scheduling and resource allocation problems in cloud systems. These methods are rule-based, problem-specific methods that seek a swift, approximate solution. Several algorithms have been extensively utilized because to their minimal processing cost and simplicity of implementation, such as First-Come-First-Serve (FCFS), Round-Robin (RR), Min-Min, and Max-Min [7][8]. Heuristics are adequate when the system is simpler or in real-time. However, in a complicated and large-scale cloud environment, it frequently fails to provide the best result.

Metaheuristic techniques are higher-level, problem-independent strategies inspired by natural processes and capable of finding near-optimal solutions by exploring vast solution spaces [9]. Due to their increased flexibility and adaptability, Complex and dynamic optimization issues in cloud environments are best suited for algorithms like Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), and Genetic Algorithm (GA). However, they frequently require careful parameter adjustment and have higher computing costs.

A. *Structured of the Paper*

The structure of this paper is as follows: Section II covers an overview of cloud resource optimization. Section III an overview of Meta-Heuristic and Heuristic Methods. Section IV outlines evaluation of Heuristic and Meta-Heuristic Methods in Comparison. Section V reviews literature and case studies, and Section VI ends with suggestions for further work.

II. CLOUD RESOURCE OPTIMIZATION: AN OVERVIEW

In cloud computing, resource optimization refers to the efficient management and utilization of computational resources to meet performance goals while reducing costs and energy usage [10]. The key resource types involved in cloud environments include in Figure 1:

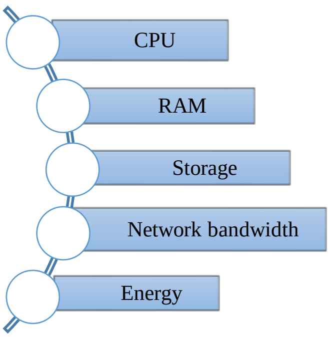

Fig. 1. Resource Types in Cloud

- **CPU (Central Processing Unit):** The main processing unit in charge of carrying out calculations and carrying out instructions. In cloud systems, virtual CPUs (vCPUs) are allocated to virtual machines (VMs) based on workload requirements. Optimizing CPU usage ensures faster task execution and prevents bottlenecks [11].
- **Memory (RAM):** Applications' actively utilized data and instructions are momentarily stored in Random Access Memory. Efficient memory management is crucial to support high-speed processing and avoid performance degradation due to paging or swapping [12].
- **Storage:** Refers to technologies for long-term data storage like SSDs, HDDs, or cloud object storage (such Amazon S3 and Azure Blob). Optimization strategies aim to reduce I/O latency, balance storage loads, and ensure high availability and scalability of data access.
- **Network Bandwidth:** The capacity to transfer data between cloud services, users, and data centers. Network resource optimization involves minimizing latency, avoiding congestion, and improving data transfer rates to support reliable and responsive services [13].
- **Energy (Power Consumption):** Cloud data centers consume substantial power to run servers, cool equipment, and maintain availability. Energy optimization focuses on reducing the overall power usage through dynamic scaling, workload consolidation, and energy-aware scheduling to support sustainable computing.

A. *Resource Allocation vs. Resource Scheduling*

Effective resource management in cloud computing requires two basic yet separate procedures: resource allocation and resource scheduling. Resource allocation is the process of allocating computing power, memory, storage, and bandwidth to virtual machines (VMs), containers, and other cloud resources or services before execution begins. Its primary objectives include maximizing resource utilization, minimizing costs, ensuring SLA compliance, and enabling elastic scalability. In contrast, resource scheduling focuses on determining when and how the allocated resources are used to execute tasks, managing execution order, prioritization, and concurrency to optimize performance indicators including energy efficiency, reaction time, and throughput [14]. While resource allocation operates at a higher level, provisioning infrastructure elements to meet application demands, scheduling operates at the task level to ensure efficient and fair use of those resources during runtime. Both processes are critical to achieving cloud optimization goals such as cost efficiency, load balancing, energy conservation, SLA adherence, and improved Quality of Service (QoS). In real-time settings, they enable cloud systems to adapt to shifting service requirements and workloads with flexibility.

B. *Applications in Dynamic Cloud Environments*

The four primary forms of cloud computing that Cloud Service Providers (CSP) provide are public, private, community, and hybrid. Creating cloud services, preserving service quality, and guaranteeing accurate distribution are the core responsibilities of CSP [15]. A lot of businesses use their own clouds only internally. The cloud computing model based on cloud service providers is shown in Figure 2:

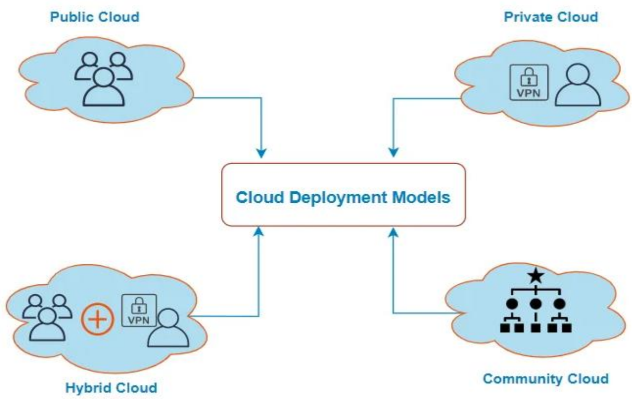

Fig. 2. Cloud Service Providers

- Private clouds- It is mostly associated with a company and is utilized only by specific consumers.
- Public clouds- In public cloud services, a third party provides the platform, infrastructure, software, and data, and consumers may access the services online. Microsoft Azure, Amazon Web Services, HP Cloud, and Google Compute Engine are a few examples [16].

- Community Cloud- It is a cloud architecture that allows a community of many organizations to use services and apps to share pertinent data.
- Hybrid cloud- It is made up of both public and private cloud resources. It makes it possible for businesses to expand their internal tools and external infrastructure. It is necessary for the cloud service provider to complete the "resource provisioning" process.

C. Challenges in Optimizing Resource Allocation in Cloud Computing Systems

There are several obstacles to maximizing resource allocation in cloud computing systems, including as [17]:

- **Resource allocation:** Decisions may involve a number of goals, including cost reduction, resource optimization, and user requirements. It might be challenging and time-consuming to optimize for many goals.
- **Optimization on a big scale:** Cloud computing systems frequently involve a vast number of users, apps, and resources. For such systems, optimizing resource allocation can be computationally demanding, requiring scalable and successful optimization techniques.
- **Dynamic optimization:** The environment of cloud computing is dynamic, with fluctuating resource availability and demands. Optimization techniques must be flexible enough to adjust in real time to such changes in order to guarantee the best possible resource allocation.
- **Heterogeneity:** A multitude of hardware and software components with varying capabilities and performance characteristics are often found in cloud computing systems [18]. Heterogeneity must be managed via optimization techniques in order to get optimal resource allocation.
- **Privacy and security:** Sensitive information like user preferences and resource availability may need to be shared in order to use optimization techniques. Such data has to be protected from hostile attacks and illegal access.
- **User satisfaction:** Optimization techniques need to include user happiness because if resources are allocated without meeting user needs, performance may suffer and users may get dissatisfied.

III. Overview of Heuristic and Metaheuristic Techniques

Rule-based methods, also known as heuristic techniques, give viable solutions to optimization problems that are complex within a reasonable time. They do not warrant a global optimal solution but are devised to find a satisfactory solution quickly through the application of domain knowledge and trivial logic [19][20]. Heuristics have been applied particularly effectively in leveraging cloud computing for task planning and resource allocation because they have low computational overhead and can be easily implemented. Such algorithms can be scenario-specific and tend to work effectively in scenario with predictable or homogeneous workloads.

A. Common Heuristic Algorithms Used

There are a number of heuristic-based scheduling algorithms widely used in cloud resources optimization namely FCFS, Round-Robin, Min-Min, Max-Min, and SJF, with each having their own strong points relating to workload and environment, as illustrated in Figure 3.

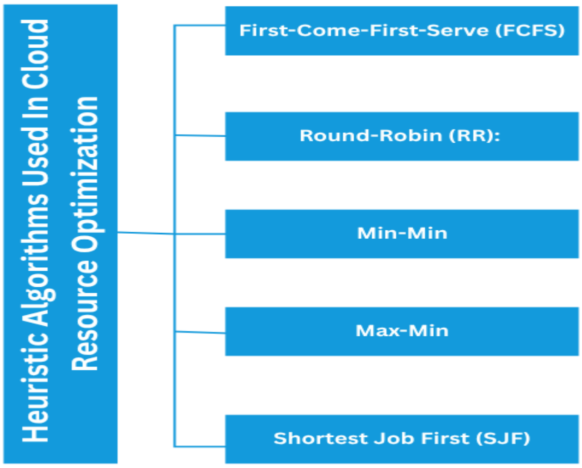

Fig. 3. Heuristic Algorithms Used in Cloud Resource Optimization

- **First-Come-First-Serve (FCFS):** First-Come, First-Served (FCFS) Regardless of their time or resource needs, FCFS schedules tasks on a first-come, first-served basis. It is easy to understand and is fair but not usually efficient [21], particularly where longer tasks impede the performance of shorter ones. FCFS may lead to inefficient resource utilization and long average wait time in cloud platforms.
- **Round-Robin (RR):** RR gives task a fixed time quantum in a cyclic manner. It is generally utilized in time-sharing surroundings. RR can produce fairness and minimize starvation in cloud resource management, however, it might not take into account task size and resource requirements and thus suboptimal schedules can be produced in a heterogeneous environment [22].
- **Min-Min:** Min-Min picks the task of least completion time in the list of tasks and places it on the resource where it can be finished first [23]. This heuristic is effective on short tasks, and reduces makespan, but can cause longer tasks to starve.
- **Max-Min:** Max-Min, as opposed to Min-Min, chooses the work with the longest anticipated completion time and assigns it to the fastest resource. It is expected to distribute the load more equally among the resources but can cause longer completion time of smaller tasks.
- **Shortest Job First (SJF):** SJF focuses on jobs that take the least time to execute. This has the advantage of reducing the average turnaround time and is more beneficial when comes to increasing throughput, which is applicable to high-performance computing in cloud environments [24]. But it presupposes the knowledge of execution time in advance and can lead to starvation of longer tasks.

B. Applications in Cloud Resource Optimization

The heuristic algorithms find extensive application in most domains of cloud resource management such as:

- **Task Scheduling**: Scheduling the tasks on the virtual machines (VMs) according to their arrival time, length, or priority with FCFS, SJF, or Min-Min.

- Virtual machines **Allocation**: VM and physical hosts are matched using heuristics to maximize resource utilization and reduce energy usage.
- **Load Balancing**: The methods such as RR are used to balance the workload to spread it equally among cloud servers to prevent overloading and unavailability of services.
- **Workflow Scheduling**: In scientific or data-intensive cloud applications, heuristics are used to manage workflow dependencies while minimizing makes pan or cost.

Although heuristics are simple and fast, their effectiveness strongly relies on a particular context and might not scale to sophisticated or changing cloud settings [25]. It is this weakness that commonly causes the need to explore more advanced metaheuristic techniques which are the subject of the next sections.

C. Fundamentals of Metaheuristic Techniques

Metaheuristic methods are optimization algorithms of high level that are based on natural phenomena like evolution, swarm intelligence, or physical annealing. Metaheuristics, unlike heuristics which are usually problem-specific, are problem-independent and can explore big, complex, and multimodal search spaces to approximate optimal solutions. They balance between exploration (global search) and exploitation (local refinement), and hence are particularly effective with NP-hard problems which include those involving cloud resource optimization [26]. They are commonly stochastic and iterative methods, in which a set of candidate solutions are refined over time according to some rules or behavior. Although they may not always get the best answer, they are very likely to produce answers that are close to ideal in a reasonable amount of computing time. They are elastic and highly adaptable and thus they fit the dynamic, heterogeneous, and scalable aspect of cloud environments well. There is some metaheuristic algorithms in cloud discuss below and in Figure 4:

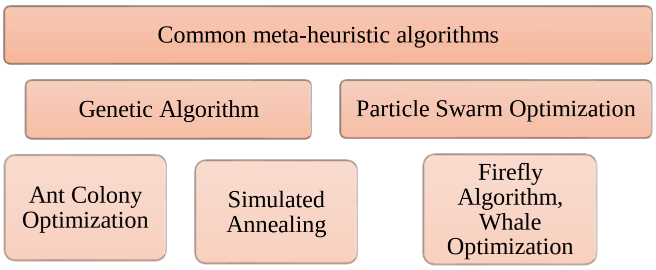

Fig. 4. Classification of Meta-Heuristic Algorithms

- **Genetic Algorithm (GA):** The principles that underpin GA are natural selection and genetics. Through crossover, mutation, and selection, an initial population of viable solutions is transformed to create generations [27]. GA has found extensive application in cloud scheduling and VM placement problems with the goal of reducing execution costs and optimizing resource utilization.
- **Particle Swarm Optimization (PSO):** PSO imitates bird or fish behavior in a social way. The search space is traversed by the particles (possible solutions) based on their best-known positions and those of their neighbors [28]. PSO has successfully applied in reducing the amount of energy used, load balancing and VM consolidation in cloud systems.
- **Ant Colony Optimization (ACO):** The foraging habits of ants, which look for the quickest path to food sources, serve as the model for ACO. Solutions built by artificial ants are founded on pheromone trails and probabilistic decision rules. In the cloud, it is used in data routing, scheduling of workflow and network optimization.
- **Simulated Annealing (SA):** SA is grounded on the annealing of metals. It accepts probability worse solutions, with the hope of escaping local optima, but slowly decreasing this probability as time goes on. SA is easy but effective in scheduling and resource allocation of small to medium cloud environments.
- **Firefly Algorithm, Whale Optimization, and Others:** These new algorithms are based on the flashing pattern of fireflies (FA) or bubble-net hunting pattern of the humpback whales (WOA). They competitively perform on multi-objective problems and have been used on real-time scheduling, deadline-conscious resource allocation, and cost-effective execution of tasks on clouds.

IV. Comparative Analysis of Heuristic and Metaheuristic Techniques

The selection of heuristic or metaheuristic method of cloud resource optimization relies on numerous performance measures and application scenarios [29]. In this section, both approaches are compared against major evaluation criteria to bring out their strengths, weaknesses, and the type of cloud environment that they are best suited for the Table I shows a brief comparison.

TABLE I. Heuristic vs. Metaheuristic Techniques – A Brief Comparison

Criterion	Heuristic Techniques	Metaheuristic Techniques
Optimization Accuracy	Provides approximate but fast solutions. Prone to local optima in complex problems.	Explores broader search space with higher chance of finding near-optimal/global solutions, especially for multi-objective and high-dimensional problems.
Scalability	Scalable in small to moderately sized systems; limited effectiveness with complex or varied workloads.	More scalable for large-scale and dynamic environments, though requires higher computation.
Convergence Speed	Fast convergence due to simple logic; suitable for real-time or near-real-time applications.	Slower convergence due to iterative search but more thorough exploration; some like PSO and SA can be tuned for speed.
Computational Overhead	Low computational cost; suitable for systems with limited resources.	Higher overhead due to population-based and iterative nature; less suitable for latency-sensitive tasks without optimization.

V. Literature Review

Based on the literature survey, Heuristic vs Metaheuristic Techniques in Cloud. It explores the challenges, innovations, and best practices related.

Ramasamy and Kamalakkannan (2025) container-based virtualization has become a cornerstone of Cloud Computing (CC) due to its lightweight and scalable properties when compared to traditional virtual machines. However, it is still difficult to optimizing resource allocation in containerized systems, and current approaches frequently fall short in addressing problems like high computational costs, scalability, cold-start scenarios, and Graphics Processing Unit (GPU) management. Container resource allocation is critical for effectively managing computing workloads in cloud settings, enabling scalability, performance optimization, and cost effectiveness. It is extensively used in a various industry, including e-commerce, healthcare, and finance, where dynamic workloads call for efficient resource management. Traditional resource allocation techniques, while their importance, frequently face difficulties such as unequal load distribution and resource underutilization [30].

Thakur, Hooda and Gill (2024) load balancing ensures equal resource allocation for optimal utilization and high user satisfaction. Recent algorithms like PSO (Particle Swarm Optimization) and ILOA (Improved Lion Optimization Algorithm) enhance Quality of Services (QoS), throughput, response time, energy consumption, and fault tolerance in cloud environments. This paper reviews load balancing techniques and parameters, showing that PSO compared with ILOA has 40% better results, than ILOA's 25%, Power Conversion's 20%, and Power Time Expense Reduction's 15% [31].

Chawla and Kaur (2024) fault-tolerant heuristic task scheduling strategy to maximize cloud computing environments' use of resources. The approach minimizes makespan and balances demand across resources by using both replication and migration strategies to achieve fault tolerance. Simulation experiments demonstrate that their algorithm achieves up to 13% improvement in resource utilization compared to benchmark algorithms under various fault-prone scenarios. The outcomes demonstrate how well the suggested algorithm works to improve resource efficiency and maintain performance in the event of resource breakdowns [32].

Zhang et al. (2023) in cloud resource scheduling, the virtual machine placement issue has become a significant obstacle. This kind of problem is commonly formulated as a vector bin packing problem and is known to be NP-hard. While heuristic methods have scalability challenges, optimization-based algorithms are unable to quickly handle on-demand customer requests for real-world large-scale problems. This study addresses the online virtual machine placement problem by presenting a VM placement model that takes into account NUMA architecture. Additionally, a method is suggested that converts optimal fine-grained solutions into coarse-grained placement policies, limiting the online implementation to heuristic placement rules [33].

Sharma and Rawat (2023) focusses on assessing and examining how well different meta-heuristic approaches work in cloud computing settings, with a specific emphasis on execution time and cost as critical performance metrics. The study begins by introducing the concept of meta-heuristic techniques, which are intelligent optimization algorithms that have gained prominence in solving complex, dynamic, and computationally expensive problems [34].

Chhabra and Basheer (2022) scheduling techniques the aim is to determine which sequence of execution of the jobs uses the least amount of processing, memory, and time. More services and very high effectiveness are usually required by the customer. The proper use of resources is facilitated by an efficient scheduling technique. In the literature, the cost and make span have been widely utilized as factors that influence the scheduling of dependent jobs. The previous study addressed performance difficulties with dependent TS (task scheduling), but not failure rate or storage cost. This research focusses on reviewing the use of heuristic and meta-heuristic techniques for cloud computing work scheduling. This paper outlines a thorough examination of TS techniques used in cloud computing, including Bee's life approach, Particle Swarm Optimization (PSO), Min-Min, Max-Min, Genetic Algorithm (GA), and others [35].

Rani, Guleria and Panda (2021) new technology that may be used for many different purposes is cloud computing. Small enterprises are drawn to cloud computing because of its cheap cost and scalability benefits, particularly in developing nations. Applications for business, data storage, backup, education, entertainment, and administration are just a few of the many uses for cloud computing. The opportunity to reach millions of consumers is enormous. Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Platform as a Service (PaaS) are the three main ways that services are delivered. This essay examines the many cloud computing principles, discusses the significance of clouds, and identifies a number of prospects [36].

The background study of cloud resource optimization, including focus study, methods, A thorough summary of the main conclusions, current restrictions, and possible future study avenues is provided in Table II.

TABLE II. SUMMARY OF LITERATURE ON HEURISTIC AND METAHEURISTIC FOR CLOUD RESOURCE OPTIMIZATION

Reference	Focus	Method	Key Findings	Limitations / Future Work
Ramasamy and Kamalakkannan (2025)	Resource allocation in container-based virtualization in cloud systems	Analytical review of containerization and resource challenges	Containerization is lightweight and scalable but struggles with cold-starts, GPU management, and uneven resource usage.	Need advanced strategies for GPU handling, cold-start mitigation, and better load balancing.
Thakur, Hooda and Gill (2024)	Cloud computing load balancing	Analyzing load balancing algorithms (such PSO, ILOA, etc.) in comparison	PSO outperforms other methods, achieving 40% improvement in QoS, throughput, and fault tolerance.	Need further testing across varied cloud architectures and workloads.
Chawla and Kaur (2024)	Fault-tolerant task scheduling	Heuristic algorithm combining replication and migration	Achieved 13% better resource utilization and balanced load under fault-prone scenarios.	Needs validation in large-scale and real-time fault-tolerant environments.
Zhang et al. (2023)	VM placement with NUMA-aware scheduling	VM placement model using coarse-grained heuristic policies from fine-grained optimization	Solves VM placement efficiently using NUMA-aware policies to address online placement challenges.	May require improvements in dynamic scaling and real-time placement decisions.

Sharma and Rawat (2023)	Evaluation of meta-heuristic methods' performance in cloud computing	Performance evaluation on execution time and cost	Meta-heuristics effectively solve complex problems with better cost-time tradeoffs.	More research needed on scalability and adaptability in dynamic workloads.
Chhabra and Basheer (2022)	Scheduling tasks with heuristic and meta-heuristic techniques	Review of scheduling algorithms, including Bee's Life, GA, PSO, Max-Min, and Min-Min	Effective scheduling can improve resource utilization and reduce makespan.	Prior studies lacked attention to failure rate and storage cost in dependent task scheduling.
Rani, Guleria and Panda (2021)	Overview of cloud computing applications and services	Conceptual review of cloud services and architecture (IaaS, PaaS, SaaS)	Cloud computing is scalable, cost-efficient, and supports various sectors including small businesses.	Does not focus on technical resource optimization or performance metrics in deployments.

VI. Conclusion and Future Work

Optimizing cloud resources remains a major challenge in modern cloud computing due to growing demands for cost-effectiveness, scalability, performance, and energy efficiency. This study reviewed fundamental cloud resources—CPU, memory, storage, bandwidth, and energy—and clarified the distinction between resource allocation and scheduling. Efficient cloud operation requires combining strategies to meet evolving user demands while ensuring SLAs and QoS. Heuristic methods like FCFS, Round-Robin, Min-Min, and SJF are quick and easy to use in static settings, but they are not very effective in dynamic, heterogeneous systems. Therefore, metaheuristics such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Simulated Annealing (SA), Ant Colony Optimization (ACO), and Whale Optimization Algorithm (WOA) are required for large-scale, multi-objective optimization.

Future research should concentrate on creating hybrid models that blend the flexibility of metaheuristics with the speed of heuristics. Energy-aware and sustainable optimization strategies must be prioritized, along with AI-driven techniques such as reinforcement learning for real-time resource management. Security, privacy, and context-awareness are critical for distributed cloud-edge systems. Furthermore, standardized datasets and benchmarks are essential for evaluating and comparing optimization approaches across varied scenarios.

References

[1] B. K. Dewangan, A. Agarwal, T. Tanupriya, and A. Pasricha, “Cloud Resource Optimization System Based on Time and Cost,” *Int. J. Math. Eng. Manag. Sci.*, vol. 5, no. 4, pp. 758–768, Aug. 2020, doi: 10.33889/IJMEMS.2020.5.4.060.

[2] S. S, R. R. Patil, and P. C, “Cloud Computing an Overview,” *Int. J. Eng. Technol.*, vol. 7, no. 4, Oct. 2018, doi: 10.14419/ijet.v7i4.10904.

[3] V. Ramamoorthi, “AI-Driven Cloud Resource Optimization Framework for Real-Time Allocation,” *J. Adv. Comput. Syst.*, vol. 1, no. 1, pp. 8–15, 2024, doi: 10.69987/JACS.2021.10102.

[4] A. R. Duggasani, “Scalable and Optimized Load Balancing in Cloud Systems: Intelligent Nature-Inspired Evolutionary Approach,” *Int. J. Innov. Sci. Res. Technol.*, vol. 10, no. 5, pp. 2153–2160, May 2025, doi: 10.38124/ijisrt/25may1290.

[5] P. Osypanka and P. Nawrocki, “Resource Usage Cost Optimization in Cloud Computing Using Machine Learning,” *IEEE Trans. Cloud Comput.*, 2022, doi: 10.1109/TCC.2020.3015769.

[6] G. Maddali, “An Efficient Bio-Inspired Optimization Framework for Scalable Task Scheduling in Cloud Computing Environments,” *Int. J. Curr. Eng. Technol.*, vol. 15, no. 3, pp. 229–238, 2025.

[7] A. Kaur and B. Kaur, “Load balancing optimization based on hybrid Heuristic-Metaheuristic techniques in cloud environment,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 3, pp. 813–824, Mar. 2022, doi: 10.1016/j.jksuci.2019.02.010.

[8] R. Patel, “Advancements in Renewable Energy Utilization for Sustainable Cloud Data Centers : A Survey of Emerging Approaches,” *Int. J. Curr. Eng. Technol.*, vol. 13, no. 5, pp. 447–454, 2023.

[9] 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*, 2021, doi: 10.1016/j.simpat.2021.102353.

[10] Y. Dongre, D. N. Ranjan, D. P. Niranjane, M. Gulhane, Y. Gandhi, and P. Karmore, “Optimizing Resource Allocation in Cloud-Based Information Systems through Machine Learning Algorithms,” *J. Inf. Syst. Eng. Manag.*, vol. 10, no. 1s, 2025.

[11] D. Patel, “The Role of AWS in Modern Cloud Architecture: Key Strategies for Scalable Deployment and Integration,” *Asian J. Comput. Sci. Eng.*, vol. 9, no. 4, 2024, doi: 10.22377/ajcse.v9i04.215.

[12] S. S. S. Neeli, “A Comparative Analysis of SQL and NoSQL Database Management within Cloud Architectures for Mission-Critical Business Systems,” *ESP Int. J. Adv. Comput. Technol.*, vol. 2, no. 4, pp. 140–149, 2024.

[13] V. S. Thokala, “Scalable Cloud Deployment and Automation for E-Commerce Platforms Using AWS, Heroku, and Ruby on Rails,” *Int. J. Adv. Res. Sci. Commun. Technol.*, pp. 349–362, Oct. 2023, doi: 10.48175/IJARSCT-13555A.

[14] V. Singh, “Reinventing Business with Cloud Integration: The Cost - Effectiveness of Replacing Legacy Applications,” *Int. J. Sci. Res.*, vol. 13, no. 8, pp. 1882–1887, 2024.

[15] J. Zhou *et al.*, “Comparative analysis of metaheuristic load balancing algorithms for efficient load balancing in cloud computing,” *J. Cloud Comput.*, 2023, doi: 10.1186/s13677-023-00453-3.

[16] A. Sharma, “Serverless Cloud Computing for Efficient Retirement Benefit Calculations,” *Int. J. Curr. Sci.*, vol. 12, no. 4, 2022.

[17] K. Kumain, “Optimization Techniques for Resource Allocation in Cloud Computing Systems,” *Turkish J. Comput. Math. Educ.*, vol. 11, no. 3, pp. 1966–1974, 2020, doi: 10.17762/turcomat.v11i3.13593.

[18] A. Mishra, “AI-Powered Cybersecurity Framework for Secure Data Transmission in IoT Network,” *Int. J. Adv. Eng. Manag.*, vol. 7, no. 3, pp. 05–13, 2025.

[19] P. Acosta-Vargas, L. A. Salvador-Ullauri, and S. Lujan-Mora, “A Heuristic Method to Evaluate Web Accessibility for Users with Low Vision,” *IEEE Access*, vol. 7, 2019, doi: 10.1109/ACCESS.2019.2939068.

[20] V. Singh, “Lessons Learned from Large-Scale Oracle Fusion Cloud Data Migrations,” *Int. J. Sci. Res.*, vol. 10, no. 10, pp. 1662–1666, 2021.

[21] M. T. D. Putra, H. Hidayat, N. Septian, and T. Afriani, “Analisis Perbandingan Algoritma Penjadwalan CPU First Come First Serve (FCFS) Dan Round Robin,” *Build. Informatics, Technol. Sci.*, vol. 3, no. 3, pp. 207–212, 2021, doi: 10.47065/bits.v3i3.1047.

[22] Sakshi *et al.*, “A new median-average round Robin scheduling algorithm: An optimal approach for reducing turnaround and waiting time,” *Alexandria Eng. J.*, vol. 61, no. 12, pp. 10527–10538, Dec. 2022, doi: 10.1016/j.aej.2022.04.006.

[23] S. M. . D. S. Mustapha and P. Gupta, “Fault aware task scheduling in cloud using min-min and DBSCAN,” *Internet Things Cyber-Physical Syst.*, vol. 4, pp. 68–76, 2024, doi: 10.1016/j.iotcps.2023.07.003.

[24] V. Sharma and A. K. Tripathi, “A systematic review of meta-heuristic algorithms in IoT based application,” *Array*, vol. 14, Jul. 2022, doi: 10.1016/j.array.2022.100164.

[25] S. B. Joseph, E. G. Dada, A. Abidemi, D. O. Oyewola, and B. M. Khammas, "Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems," *Heliyon*, vol. 8, no. 5, p. e09399, May 2022, doi: 10.1016/j.heliyon.2022.e09399.

[26] P. Li, H. Wang, G. Tian, and Z. Fan, "Towards Sustainable Cloud Computing: Load Balancing with Nature-Inspired Meta-Heuristic Algorithms," *Electronics*, vol. 13, no. 13, p. 2578, Jun. 2024, doi: 10.3390/electronics13132578.

[27] H. Zhang, "An immune genetic algorithm for simple assembly line balancing problem of type 1," *Assem. Autom.*, vol. 39, no. 1, pp. 113–123, Apr. 2019, doi: 10.1108/AA-08-2017-101.

[28] M. Masrom, N. A Ghani, and M. Tokhi, "Particle swarm optimization and spiral dynamic algorithm-based interval type-2 fuzzy logic control of triple-link inverted pendulum system: A comparative assessment," *J. Low Freq. Noise, Vib. Act. Control*, vol. 40, no. 1, pp. 367–382, Mar. 2021, doi: 10.1177/1461348419873780.

[29] G. F. Shidik, E. J. Kusuma, S. Nuraisha, and P. N. Andono, "Heuristic vs Metaheuristic Method: Improvement of Spoofed Fingerprint Identification in IoT Devices," *Int. Rev. Model. Simulations*, vol. 12, no. 3, Jun. 2019, doi: 10.15866/iremos.v12i3.17330.

[30] G. Ramasamy and S. Kamalakkannan, "Enhancement of Containerization-based Cloud Computing Performance using Hybrid Metaheuristic Optimization Algorithm: A Review," in *International Conference on Electronics and Renewable Systems (ICEARS)*, 2025, pp. 1072–1079. doi: 10.1109/ICEARS64219.2025.10941481.

[31] M. Thakur, S. Hooda, and R. Gill, "Optimizing Resource Allocation using Metaheuristic Approach for Load Balancing in Cloud Computing," in *2024 4th International Conference on Sustainable Expert Systems (ICSES)*, 2024, pp. 559–564. doi: 10.1109/ICSES63445.2024.10763064.

[32] S. Chawla and A. Kaur, "Fault-Tolerant Heuristic Task Scheduling Algorithm for Efficient Resource Utilization in Cloud Computing," in *2024 International Conference on Automation and Computation (AUTOCOM)*, 2024, pp. 131–135. doi: 10.1109/AUTOCOM60220.2024.10486076.

[33] Y. Zhang *et al.*, "Generating Heuristic Policies from Optimization in Large Scale Cloud Computing VM scheduling," in *2023 42nd Chinese Control Conference (CCC)*, IEEE, Jul. 2023, pp. 2013–2020. doi: 10.23919/CCC58697.2023.10240954.

[34] S. Sharma and P. S. Rawat, "Performance Evaluation and Analysis of Meta-Heuristic Techniques in Cloud Computing," in *2023 10th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, IEEE, Dec. 2023, pp. 1254–1258. doi: 10.1109/UPCON59197.2023.10434873.

[35] M. Chhabra and S. Basheer, "Recent Task Scheduling-based Heuristic and Meta-heuristics Methods in Cloud Computing: A Review," in *2022 5th International Conference on Contemporary Computing and Informatics (IC3I)*, IEEE, Dec. 2022, pp. 2236–2242. doi: 10.1109/IC3I56241.2022.10073445.

[36] M. Rani, K. Guleria, and S. N. Panda, "Cloud Computing An Empowering Technology: Architecture, Applications and Challenges," in *2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO)*, IEEE, Sep. 2021, pp. 1–6. doi: 10.1109/ICRITO51393.2021.9596259.

