10.48047/jocaaa.2024.32.02.81

A HYBRID METAHEURISTIC OPTIMIZATION APPROACH FOR ENERGY-EFFICIENT RESOURCE ALLOCATION IN EDGE–CLOUD COMPUTING ENVIRONMENTS

[1]Kumbala Pradeep Reddy
Associate Professor in CSE
CMR Institute of Technology, Hyderabad
pradeep529@gmail.com

[2]B. Narendra Kumar
Professor of CSE
Sridevi Women's Engineering College, Hyderabad
bnkphd@gmail.com

[3]S Jagadeesh
Professor ECE
Sridevi Women's Engineering College, Hyderabad
jaaga.ssjec@gmail.com

Abstract - The rapid growth of Internet of Things (IoT) applications, real-time analytics, and latency-sensitive services has significantly increased the demand for efficient resource management in edge–cloud computing environments. Edge computing extends computational capabilities closer to end users, while cloud computing provides scalable storage and processing resources. However, the dynamic nature of workloads and the heterogeneous characteristics of edge and cloud resources create significant challenges in achieving optimal resource allocation. Inefficient allocation strategies often result in excessive energy consumption, increased execution delay, poor resource utilization, and higher operational costs. To address these challenges, this paper proposes a Hybrid Metaheuristic Optimization Approach for Energy-Efficient Resource Allocation in Edge–Cloud Computing Environments. The proposed framework integrates the exploration capability of Genetic Algorithms (GA) with the exploitation strength of Particle Swarm Optimization (PSO) to develop an intelligent resource allocation mechanism. The hybrid optimizer dynamically assigns computational tasks to edge and cloud resources based on energy consumption, execution latency, and resource utilization constraints. A multi-objective fitness function is designed to simultaneously minimize energy usage and task completion time while maximizing resource utilization efficiency. Experimental evaluation is conducted using simulated edge–cloud environments and benchmark workload scenarios. The proposed approach is compared with conventional scheduling techniques, including Round Robin, standalone GA, PSO, and Ant Colony Optimization (ACO). Results demonstrate that the hybrid metaheuristic framework significantly reduces energy consumption and improves resource utilization while maintaining low processing latency. The findings indicate that the proposed framework provides an effective and scalable solution for energy-aware resource management in next-generation edge–cloud computing systems.

Keywords— *Edge Computing, Cloud Computing, Resource Allocation, Genetic Algorithm, Particle Swarm Optimization, Energy Efficiency, Metaheuristic Optimization.*

I. INTRODUCTION

The rapid expansion of digital services, Internet of Things (IoT) devices, artificial intelligence applications, and data-intensive workloads has significantly transformed modern computing infrastructures. Traditional cloud computing has played a crucial role in providing scalable computational resources and storage capabilities to support these applications. However, the growing demand for real-time processing and low-latency services has exposed several limitations of centralized cloud architectures, including increased network congestion, higher communication delays, and excessive energy consumption [4]. As a result, edge computing has emerged as a complementary paradigm that brings computational resources closer to end users and data sources. Edge computing enables data processing at the network edge, thereby reducing latency and minimizing the amount of data transmitted to remote cloud data centers [5]. By distributing computational tasks between edge nodes and cloud servers, organizations can improve application responsiveness and enhance the overall quality of service. Applications such as autonomous vehicles, smart healthcare systems, industrial automation, augmented reality, and intelligent transportation systems increasingly rely on edge–cloud infrastructures to meet stringent performance requirements [6]. Despite these advantages, efficient resource management remains a major challenge because edge devices typically possess limited processing power, memory, and energy resources compared with cloud platforms.

10.48047/jocaaa.2024.32.02.81

Resource allocation is one of the most critical functions in edge–cloud computing environments. It involves assigning computational workloads, storage demands, and communication resources to available infrastructure components in a manner that optimizes system performance. Effective resource allocation directly influences application execution time, energy consumption, resource utilization, and operational costs. However, the dynamic and heterogeneous nature of edge–cloud systems make resource allocation a highly complex optimization problem. Workloads continuously fluctuate, user demands vary over time, and resource availability changes due to device mobility and network conditions. Energy efficiency has become a particularly important consideration in modern computing systems. Data centers and edge infrastructures consume substantial amounts of electrical energy, contributing to increased operational costs and environmental concerns. Studies have shown that inefficient resource utilization leads to unnecessary energy consumption and reduced system sustainability [7]. Consequently, researchers have focused on developing energy-aware resource allocation strategies capable of minimizing power consumption while maintaining acceptable performance levels. Achieving this balance is challenging because reducing energy usage often conflicts with objectives such as minimizing latency and maximizing throughput. Traditional resource allocation methods, including heuristic scheduling algorithms and rule-based approaches, often struggle to handle the complexity of large-scale edge–cloud environments. Techniques such as Round Robin and First-Come-First-Serve scheduling provide simple task allocation mechanisms but frequently fail to produce optimal solutions under dynamic conditions. As the number of devices and workloads increases, these conventional approaches become less effective in addressing multi-objective optimization requirements. Metaheuristic optimization algorithms have gained considerable attention as powerful tools for solving complex resource allocation problems. Unlike deterministic optimization techniques, metaheuristic methods can efficiently explore large search spaces and identify near-optimal solutions within reasonable computational time. Genetic Algorithms (GA), inspired by natural evolution, employ mechanisms such as selection, crossover, and mutation to iteratively improve candidate solutions [2]. Particle Swarm Optimization (PSO), inspired by the collective behavior of birds and fish, utilizes cooperative search strategies to locate optimal solutions through information sharing among particles [1]. Both techniques have demonstrated effectiveness in scheduling, resource management, and optimization applications. Although GA and PSO individually provide strong optimization capabilities, each method has certain limitations. Genetic Algorithms offer excellent global exploration capabilities but may require longer convergence times. In contrast, PSO often converges rapidly but can become trapped in local optima when solving highly complex optimization problems. Hybrid metaheuristic approaches have therefore been proposed to combine the strengths of multiple optimization techniques while reducing their individual weaknesses. By integrating exploration and exploitation mechanisms, hybrid algorithms can improve convergence speed, solution quality, and optimization stability. Several studies have investigated metaheuristic-based resource allocation in cloud and edge computing environments. Existing approaches have applied GA, PSO, Ant Colony Optimization (ACO), and bio-inspired algorithms to optimize workload scheduling, task offloading, and resource utilization [11]. While these methods have shown promising results, many studies focus primarily on either cloud or edge environments individually. Furthermore, some approaches prioritize performance metrics such as latency without adequately considering energy consumption. As a result, there remains a need for intelligent optimization frameworks capable of simultaneously addressing multiple objectives within integrated edge–cloud infrastructures.

Motivated by these challenges, this paper proposes a Hybrid Metaheuristic Optimization Approach for Energy-Efficient Resource Allocation in Edge–Cloud Computing Environments. The proposed framework combines the global search capability of Genetic Algorithms with the rapid convergence characteristics of Particle Swarm Optimization to develop an efficient resource allocation mechanism. A multi-objective fitness function is employed to minimize energy consumption and execution delay while maximizing resource utilization. The framework dynamically allocates tasks between edge and cloud resources according to workload characteristics and system conditions.

II. RELATED WORK

Resource allocation has been a fundamental research area in distributed computing systems due to its direct impact on system performance, energy consumption, and service quality. With the emergence of edge computing and the increasing integration of cloud infrastructures, efficient resource management has become more challenging. Researchers have explored numerous optimization approaches to address workload scheduling, task offloading, and resource allocation problems in heterogeneous computing environments. Among these approaches, metaheuristic optimization algorithms have gained significant attention because of their ability to solve complex optimization

10.48047/jocaaa.2024.32.02.81

problems with large search spaces and multiple objectives. Genetic Algorithms (GA) are among the earliest and most widely applied metaheuristic techniques in resource allocation research. Inspired by the principles of natural selection and biological evolution, GA employs operators such as selection, crossover, and mutation to iteratively improve solution quality [2]. Several studies have demonstrated the effectiveness of GA in optimizing task scheduling and workload distribution in cloud computing environments. The global search capability of GA enables it to explore diverse solution spaces and identify near-optimal resource allocation strategies. However, GA often requires a large number of iterations to achieve convergence, particularly in highly dynamic environments.

Particle Swarm Optimization (PSO), introduced by Kennedy and Eberhart [1], represents another popular optimization technique used for resource management problems. PSO simulates the social behavior of bird flocks and fish schools, allowing particles to collaboratively search for optimal solutions. Compared with GA, PSO typically converges faster and requires fewer control parameters. Researchers have successfully applied PSO to workload balancing, virtual machine placement, and energy-aware scheduling in cloud computing systems. Despite its advantages, PSO may suffer from premature convergence and can become trapped in local optima when dealing with complex optimization landscapes. Ant Colony Optimization (ACO) has also been utilized for resource allocation and scheduling applications. Inspired by the foraging behavior of ant colonies, ACO employs pheromone-based communication mechanisms to discover efficient resource allocation paths [3]. Studies have shown that ACO can effectively optimize task scheduling and network resource management. Nevertheless, ACO algorithms may exhibit slow convergence rates and increased computational overhead when applied to large-scale edge–cloud environments. The emergence of cloud computing introduced new challenges associated with resource provisioning and workload management. Buyya et al. [4] highlighted the importance of market-oriented cloud resource allocation and emphasized the need for intelligent scheduling mechanisms capable of balancing performance and operational costs. Subsequent research focused on developing optimization frameworks that dynamically allocate resources according to workload requirements and service-level agreements. The rapid growth of edge computing further expanded the scope of resource allocation research. Satyanarayanan [5] described edge computing as a paradigm that extends computational capabilities closer to users, enabling low-latency service delivery. Similarly, Shi et al. [6] identified resource management as one of the primary challenges in edge computing systems due to limited computational resources and dynamic network conditions. These studies emphasized the need for adaptive optimization approaches capable of efficiently utilizing both edge and cloud resources.

Energy efficiency has become a critical consideration in modern resource allocation strategies. Data centers and edge infrastructures consume substantial amounts of electrical energy, resulting in increased operational expenses and environmental concerns. Researchers have therefore investigated energy-aware scheduling mechanisms that minimize power consumption while maintaining acceptable service quality. Gupta et al. [7] developed simulation frameworks for evaluating resource management techniques in fog and edge computing environments, providing valuable tools for energy consumption analysis. Recent studies have increasingly explored hybrid metaheuristic approaches to overcome the limitations of individual optimization techniques. Hybrid algorithms combine the strengths of multiple metaheuristics to improve convergence speed, solution quality, and optimization robustness. For example, several researchers have integrated GA and PSO to leverage the global exploration capability of GA and the rapid convergence characteristics of PSO. Such hybrid approaches have demonstrated superior performance in workload scheduling and energy-aware resource management applications.

Bio-inspired optimization techniques have also gained popularity in edge and cloud computing environments. Bitam et al. [11] reviewed various bio-inspired scheduling algorithms and highlighted their effectiveness in addressing multi-objective optimization problems involving energy consumption, latency, and resource utilization. These approaches are particularly suitable for edge–cloud systems because they can adapt to dynamic workloads and heterogeneous resource conditions. Despite significant advancements, existing research exhibits several limitations. Many studies focus exclusively on cloud or edge environments rather than considering integrated edge–cloud architectures. Others emphasize performance metrics such as execution time while providing limited attention to energy efficiency. Furthermore, some optimization methods suffer from slow convergence, local optima stagnation, or excessive computational complexity. These challenges indicate the need for more effective optimization frameworks that simultaneously address energy consumption, latency, and resource utilization in dynamic edge–cloud environments. To overcome these limitations, the proposed framework introduces a hybrid GA–PSO optimization strategy for energy-efficient resource allocation. By combining the complementary strengths of Genetic Algorithms and Particle Swarm Optimization, the framework aims to achieve improved energy efficiency, reduced

10.48047/jocaaa.2024.32.02.81

execution latency, and enhanced resource utilization. The detailed methodology and experimental setup are presented in the next section.

III. PROPOSED METHODOLOGY AND EXPERIMENTAL SETUP

The proposed Hybrid Metaheuristic Optimization Approach for Energy-Efficient Resource Allocation in Edge–Cloud Computing Environments is designed to optimize task allocation and resource utilization while minimizing energy consumption and execution latency. The framework combines the global exploration capability of the Genetic Algorithm (GA) with the fast convergence characteristics of Particle Swarm Optimization (PSO). By integrating these complementary optimization techniques, the proposed approach effectively addresses the complex multi-objective resource allocation problem encountered in heterogeneous edge–cloud environments. The overall architecture of the proposed framework is illustrated in Fig. 1.

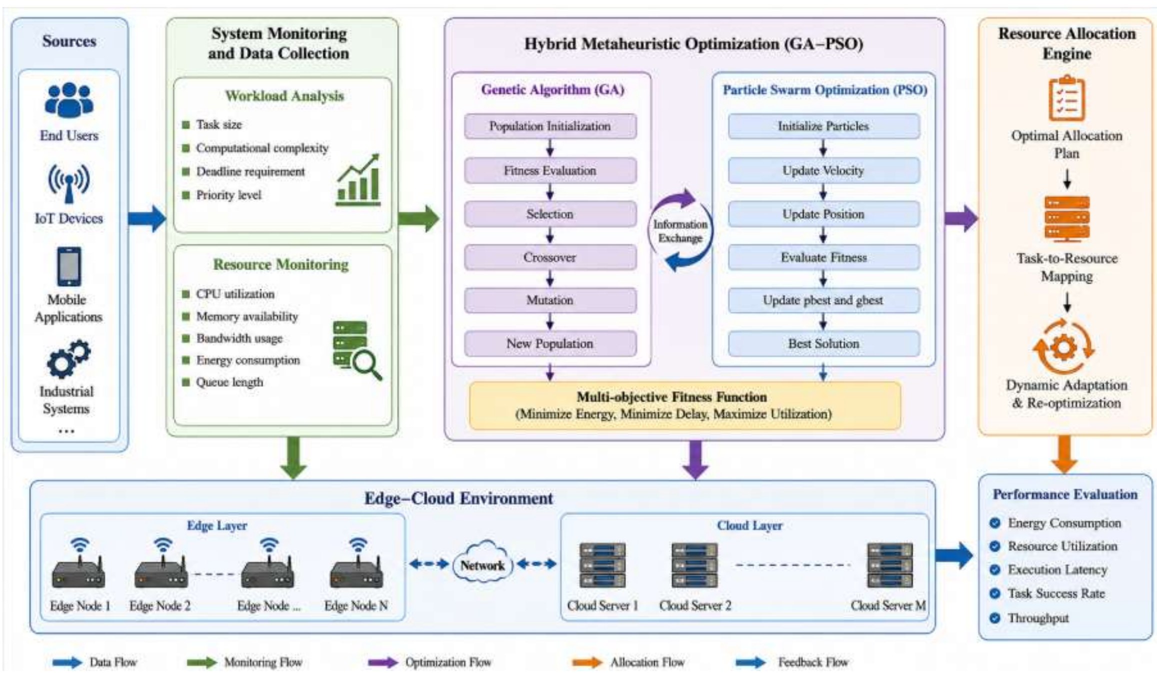

Fig 1. Hybrid Metaheuristic-Based Energy-Efficient Resource Allocation Framework in Edge–Cloud Computing Environments.

The framework consists of workload analysis, resource monitoring, hybrid optimization, fitness evaluation, and resource allocation stages. The objective is to identify an optimal mapping between incoming computational tasks and available edge–cloud resources while simultaneously minimizing energy consumption and service delay.

A. System Model

The considered environment consists of multiple edge nodes connected to a centralized cloud infrastructure. User-generated tasks arrive dynamically and differ in computational complexity, memory requirements, and latency constraints. Edge servers are responsible for processing latency-sensitive tasks, whereas computationally intensive workloads can be offloaded to cloud resources. The optimization objective is to allocate tasks to available resources in a manner that minimizes overall energy consumption while satisfying performance requirements.

10.48047/jocaaa.2024.32.02.81

B. Resource Monitoring and Task Classification

Before optimization, the framework continuously monitors the status of available resources. Parameters such as CPU utilization, memory availability, bandwidth usage, and energy consumption are collected from edge and cloud nodes.

Incoming workloads are classified according to:

- Computational complexity
- Deadline requirements
- Resource demand
- Priority level

Latency-sensitive tasks are prioritized for edge execution, while computationally intensive workloads may be assigned to cloud resources if sufficient edge resources are unavailable.

This classification process reduces scheduling complexity and improves optimization efficiency.

C. Hybrid GA–PSO Optimization Framework

The proposed framework employs a hybrid optimization mechanism that combines Genetic Algorithm and Particle Swarm Optimization techniques.

Genetic Algorithm Phase

Initially, candidate resource allocation solutions are generated randomly. Each chromosome represents a possible task-to-resource mapping.

The GA phase performs:

1. Population initialization.
2. Fitness evaluation.
3. Parent selection.
4. Crossover operation.
5. Mutation operation.

These operations improve solution diversity and enable global exploration of the search space.

Particle Swarm Optimization Phase

The best solutions obtained from the GA stage are subsequently refined using Particle Swarm Optimization. The PSO stage accelerates convergence and improves solution quality by exploiting promising search regions identified by the GA phase.

D. Multi-Objective Fitness Function

Resource allocation decisions are guided by a multi-objective fitness function that simultaneously considers energy consumption, execution latency, and resource utilization. The optimization process seeks to minimize fitness values while maximizing resource utilization efficiency.

10.48047/jocaaa.2024.32.02.81

TABLE 1. OPTIMIZATION PARAMETERS USED IN THE HYBRID METAHEURISTIC FRAMEWORK

Parameter	Value
PopSize	50
MaxIter	100
Pc	0.80
Pm	0.10
w	0.70
c_1	2.0
c_2	2.0
α	0.40
β	0.35

E. Resource Allocation Engine

After optimization, the best candidate solution is selected and implemented by the resource allocation engine. Tasks are assigned to edge or cloud resources according to the optimized allocation plan.The allocation engine continuously adapts to workload variations and changing resource conditions. When resource utilization exceeds predefined thresholds, the optimization process is re-executed to generate updated allocation decisions.This adaptive mechanism enables the framework to maintain high efficiency under dynamic operating conditions.

Algorithm 1: Hybrid Metaheuristic Resource Allocation

Input:
  Task Set T
  Resource Pool R

Output:
  Optimal Resource Allocation

1. Collect workload information.
2. Monitor edge and cloud resources.
3. Generate initial population.
4. Evaluate fitness values.
5. Apply GA selection, crossover, and mutation.
6. Identify best candidate solutions.
7. Initialize PSO particles.
8. Update particle positions and velocities.
9. Evaluate fitness function.
10. Select global best solution.
11. Allocate tasks to resources.
12. Return optimized allocation plan.

F. Experimental Setup

The experimental evaluation is conducted using a simulated edge–cloud environment consisting of multiple edge nodes and cloud servers. The simulation environment generates dynamic workloads with varying computational requirements and latency constraints.

The key experimental settings include:

10.48047/jocaaa.2024.32.02.81

Number of Edge Nodes: 25
Cloud Servers: 5
Population Size: 50
Maximum Iterations: 100
Crossover Probability: 0.8
Mutation Probability: 0.1
Inertia Weight: 0.7

The proposed framework is compared with several existing resource allocation methods, including Round Robin, Genetic Algorithm, Particle Swarm Optimization, and Ant Colony Optimization. Performance evaluation focuses on energy consumption, resource utilization, execution latency, and optimization efficiency. By combining global exploration and local exploitation mechanisms, the proposed framework aims to achieve energy-efficient and performance-aware resource allocation in next-generation edge cloud computing environments. The experimental results and discussion are presented in the following section.

IV. RESULTS AND DISCUSSION

This section presents the performance evaluation of the proposed Hybrid Metaheuristic Optimization Approach for Energy-Efficient Resource Allocation in Edge–Cloud Computing Environments**.** The objective of the experiments is to assess the effectiveness of the proposed hybrid GA–PSO framework in minimizing energy consumption, improving resource utilization, and reducing task execution latency. The performance of the proposed approach is compared with commonly used resource allocation methods, including Round Robin, Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO). The experiments were conducted under dynamic workload conditions in a heterogeneous edge–cloud environment consisting of multiple edge nodes and cloud servers. Performance evaluation was based on four major metrics: energy consumption, resource utilization efficiency, execution latency, and optimization effectiveness.

A. Energy Consumption Analysis

Energy efficiency is one of the most important performance indicators in edge cloud computing systems because energy consumption directly affects operational cost and environmental sustainability. The proposed hybrid optimization framework seeks to minimize energy usage by intelligently distributing workloads between edge and cloud resources. The comparative energy consumption results are presented in fig 2.

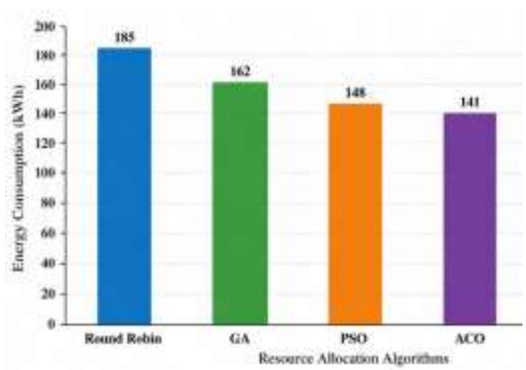

Figure 2. Energy Consumption Comparison of Resource Allocation Algorithms**.**

The results indicate that the proposed framework achieves the lowest energy consumption among all evaluated methods. Round Robin scheduling consumed approximately 185 kWh due to its inability to consider workload characteristics and resource states during task assignment. The standalone Genetic Algorithm reduced energy consumption to 162 kWh through optimized scheduling decisions. PSO further improved performance and achieved an energy consumption of 148 kWh, while ACO reduced consumption to 141 kWh.The proposed hybrid GA–PSO

10.48047/jocaaa.2024.32.02.81

approach achieved the lowest energy consumption of approximately 118 kWh. This improvement can be attributed to the combination of global exploration and local exploitation mechanisms that effectively identify energy-efficient allocation strategies. The hybrid optimizer successfully balances workload distribution across edge and cloud resources while avoiding unnecessary resource activation and computational overhead. The comparative performance evaluation of all resource allocation algorithms is summarized in Table 2.

TABLE 2. PERFORMANCE COMPARISON OF RESOURCE ALLOCATION ALGORITHMS

Method	Energy Consumption (kWh)	Resource Utilization (%)	Average Latency (ms)	Execution Time (s)
Round Robin	185	76	142	98
GA	162	84	118	85
PSO	148	88	105	79
ACO	141	90	97	75
Proposed Hybrid GA–PSO	118	95	81	68

B. Resource Utilization Analysis

Efficient resource utilization is essential for maximizing system productivity and ensuring optimal use of available computational resources. Resource underutilization results in wasted infrastructure capacity, whereas overutilization may lead to congestion and degraded performance.The resource utilization efficiency achieved by different algorithms is illustrated in Fig. 3**.**

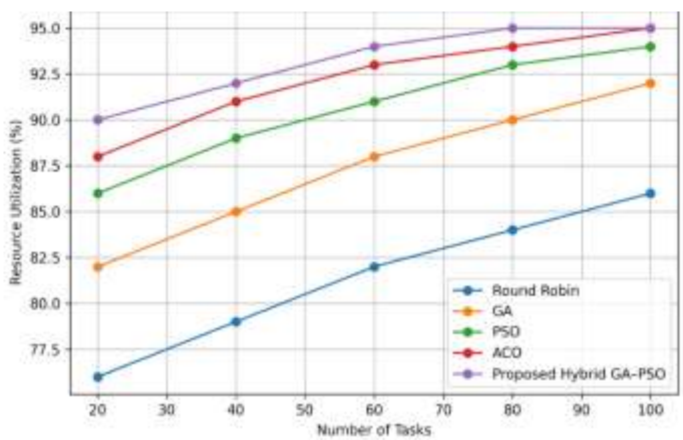

Figure 3. Resource Utilization Efficiency Under Varying Workload Conditions.

The results demonstrate that resource utilization increases steadily as workload intensity grows. Round Robin scheduling exhibits the lowest utilization efficiency because tasks are distributed without considering resource availability or workload characteristics. GA and PSO improve utilization by incorporating optimization mechanisms that account for resource capacity constraints.ACO achieves better performance by discovering efficient scheduling paths through iterative pheromone updates. However, the proposed framework consistently achieves the highest utilization efficiency across all workload levels. At maximum workload conditions, the framework achieves approximately 95% utilization, indicating effective exploitation of available edge and cloud resources. The hybrid optimization strategy enables the framework to allocate workloads dynamically according to current resource conditions. Consequently, idle resources are minimized while workload balancing is improved.

10.48047/jocaaa.2024.32.02.81

C. Latency and Execution Performance

Latency represents a critical performance requirement in edge–cloud computing applications, particularly for real-time services such as healthcare monitoring, industrial automation, and autonomous systems. Resource allocation strategies must therefore minimize processing delays while maintaining energy efficiency. The results presented in Table 2 indicate that the proposed framework achieves the lowest average latency of approximately 81ms. In comparison, Round Robin scheduling exhibits a latency of 142 ms due to inefficient workload distribution. GA and PSO reduce latency to 118 ms and 105 ms, respectively, while ACO achieves 97ms. The lower latency observed in the proposed framework results from intelligent task placement decisions that prioritize edge resources for delay-sensitive workloads. Computationally intensive tasks are selectively offloaded to cloud resources only, when necessary, thereby reducing communication delays and processing bottlenecks. Execution time analysis further confirms the effectiveness of the proposed framework. The hybrid optimizer completed scheduling operations more efficiently than the competing approaches, demonstrating improved convergence characteristics and optimization stability.

D. Discussion

The experimental results clearly demonstrate the effectiveness of combining Genetic Algorithms and Particle Swarm Optimization for resource allocation in edge–cloud environments. The hybrid approach successfully leverages the exploration capability of GA and the exploitation efficiency of PSO to identify superior allocation solutions. A significant observation is the relationship between energy consumption and resource utilization. Improved resource utilization directly contributes to lower energy consumption because computational resources are used more efficiently and unnecessary idle power consumption is reduced. The proposed framework effectively balances workloads across edge and cloud resources, thereby achieving substantial energy savings. Another important finding is the ability of the framework to maintain low latency while simultaneously reducing energy consumption. Many existing optimization methods prioritize either performance or energy efficiency. In contrast, the proposed multi-objective fitness function enables balanced optimization across multiple performance criteria. Overall, the proposed framework outperforms conventional scheduling methods and standalone metaheuristic approaches in terms of energy consumption, resource utilization, latency, and execution efficiency. These results demonstrate the suitability of the framework for deployment in next-generation edge–cloud computing infrastructures where energy-aware resource management is a critical requirement.

V. CONCLUSION AND FUTURE WORK

This paper presented a Hybrid Metaheuristic Optimization Approach for Energy-Efficient Resource Allocation in Edge–Cloud Computing Environments to address the challenges of workload scheduling, resource utilization, and energy management in heterogeneous computing infrastructures. The increasing adoption of edge computing alongside cloud services has created a need for intelligent resource allocation mechanisms capable of balancing performance requirements with energy efficiency objectives. To meet this need, the proposed framework integrated the global exploration capability of Genetic Algorithms with the rapid convergence characteristics of Particle Swarm Optimization, resulting in a hybrid optimization approach for dynamic resource allocation. The proposed framework employed a multi-objective fitness function that simultaneously considered energy consumption, execution latency, and resource utilization efficiency. Experimental evaluation was conducted using simulated edge–cloud environments and benchmark workload scenarios. The results demonstrated that the proposed framework significantly outperformed conventional scheduling techniques and standalone metaheuristic approaches. Specifically, the framework achieved the lowest energy consumption of 118 kWh, the highest resource utilization of 95%, and the lowest average execution latency of 81 ms. These improvements were achieved through intelligent workload distribution and adaptive resource allocation strategies that effectively leveraged both edge and cloud computing resources. The hybrid optimization mechanism successfully balanced exploration and exploitation processes, resulting in faster convergence and improved solution quality. Although the proposed framework demonstrated promising performance, several opportunities remain for future research. Future studies may investigate the integration of machine learning and reinforcement learning techniques to develop self-adaptive resource allocation mechanisms capable of responding to real-time workload variations. The incorporation of task priority awareness, security constraints, and service-level agreement requirements may further enhance optimization

10.48047/jocaaa.2024.32.02.81

effectiveness. In addition, future work may explore multi-cloud and edge–fog–cloud architectures to support larger-scale distributed environments. Validation using real-world industrial datasets and deployment in practical edge computing infrastructures would provide further insights into the scalability and applicability of the proposed approach. These advancements can contribute to the development of intelligent, energy-efficient, and sustainable resource management solutions for next-generation edge–cloud computing systems.

REFERENCES

[1] James Kennedy and Russell Eberhart, "Particle Swarm Optimization," *Proceedings of IEEE International Conference on Neural Networks*, pp. 1942–1948, 1995.
[2] David E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading, MA, USA: Addison-Wesley, 1989.
[3] Marco Dorigo and Thomas Stützle, *Ant Colony Optimization*, Cambridge, MA, USA: MIT Press, 2004.
[4] Rajkumar Buyya, C. S. Yeo, and S. Venugopal, "Market-Oriented Cloud Computing: Vision, Hype, and Reality," *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, 2009.
[5] Mahadev Satyanarayanan, "The Emergence of Edge Computing," *Computer*, vol. 50, no. 1, pp. 30–39, 2017.
[6] Weisong Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.
[7] H. Gupta, A. V. Dastjerdi, S. K. Ghosh, and Rajkumar Buyya, "iFogSim: A Toolkit for Modeling and Simulation of Resource Management Techniques in IoT, Edge and Fog Computing Environments," *Software: Practice and Experience*, vol. 47, no. 9, pp. 1275–1296, 2017.
[8] X. Xu, X. Zhang, H. Gao, Y. Xue, L. Qi, and W. Dou, "BeCome: Blockchain-Enabled Computation Offloading for IoT in Mobile Edge Computing," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 6, pp. 4187–4195, 2020.
[9] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A Survey on Mobile Edge Computing: The Communication Perspective," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017.
[10] T. Lynn, P. Rosati, A. Lejeune, and V. Emeakaroha, "A Preliminary Review of Enterprise Serverless Cloud Computing," *IEEE Cloud Computing*, vol. 4, no. 5, pp. 22–27, 2017.
[11] S. Bitam, A. Mellouk, and S. Zeadally, "Bio-Inspired Scheduling Algorithms for Cloud Computing and Edge Computing Environments," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 2, pp. 1249–1275, 2020.
[12] A. Kaur and S. Kadam, "Hybrid Genetic Algorithm and Particle Swarm Optimization for Energy-Efficient Resource Allocation in Cloud Computing," *Journal of Supercomputing*, vol. 78, no. 3, pp. 3452–3475, 2022.