International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING

ISSN:2147-6799 www.ijisae.org Original Research Paper

Optimizing Cloud Resource Allocation and Load Balancing through Eco-Efficient Task Scheduling

S. Amutha[1], P. Kamarajapandian[2], J. Nirmaladevi[3], S. Saravanan[4], S. Vijayalakshmi[5], S. Athimoolam[6]

Submitted: 03/11/2023 **Revised**: 22/12/2023 **Accepted**: 04/01/2024

Abstract: Cloud computing is a rapidly evolving field that requires efficient resource allocation and fair distribution of tasks to achieve optimal performance and cost-effectiveness. To address these concerns, this study explores EcoSched, a pioneering study in the realm of cloud computing, aiming to revolutionize resource allocation and task scheduling methodologies for enhanced efficiency and sustainability. In response to the evolving demands of this field, this research investigates dynamic task scheduling methods tailored to optimize resource utilization and task distribution in cloud environments. This innovative framework emphasizes the eco-efficient assignment of tasks by categorizing them based on computational intensity, interdependencies, and stringent deadlines. Employing a refined task assignment mechanism supported by a sophisticated dynamic task scheduler, tasks are intelligently allocated to suitable virtual machines in real-time. Moreover, the integration of heuristic and predictive analysis enhances the decision-making process within the scheduler, ensuring optimal task placement. In parallel, EcoSched incorporates a robust load balancer capable of dynamically adjusting task allocations across the cloud infrastructure. By proactively mitigating resource bottlenecks and minimizing response times, this load balancer significantly enhances system performance. The proposed methodology showcases remarkable improvements in response time and resource utilization metrics, surpassing conventional scheduling approaches. This research offers valuable insights into the scalability and adaptability of the introduced techniques, laying the groundwork for future advancements in dynamic task scheduling strategies. With a focus on optimizing resource allocation and load balancing, this study contributes to the evolution of resilient, efficient, and sustainable cloud environments. EcoSched sets the stage for meeting the escalating computational demands while promoting eco-efficiency, thus shaping the future landscape of cloud computing.

Keywords: *Cloud computing, EcoShed, Task Scheduling, Resource Allocation, Load Balancing, Dynamic Task Scheduler, Computational Efficiency*

1. Introduction

Cloud computing has completely transformed the way organizations manage and utilize their resources, ushering in a new era of computational capabilities. The flexibility, scalability, and cost-efficiency offered by cloud services make it the backbone of modern IT systems. However, as the reliance on cloud services continues to grow, the challenge of effectively allocating resources and balancing workloads has become increasingly prominent [1].

Traditional static task scheduling methods, designed for more predictable conditions, struggle to adapt to the dynamic and heterogeneous nature of today's cloud computing environments. This often leads to underutilization of resources and performance bottlenecks, highlighting the need for dynamic task scheduling techniques that can respond in real-time to changing workloads. Managing a large number of tasks spread across a distributed network of virtual machines (VMs) in the cloud infrastructure is a complex task. These tasks have varying computational requirements, dependencies, and time constraints, making the task scheduling problem inherently challenging. Cloud workloads are dynamic, with fluctuations in demand and the constant emergence of new tasks, making intelligent task allocation crucial [2]. The traditional one-size-fits-all approach is not suitable for dealing with such complexities, leading to suboptimal allocation of resources and lack of responsiveness in the rapidly-evolving cloud computing landscape.

This study provides a comprehensive framework for dynamic project scheduling that addresses the challenges posed by cloud computing environments. The objectives are threefold: firstly, to define tasks based on their

[1]*Associate Professor, Department of Computer Science and Engineering, School of Computing, Vel Tech Rangarajan Dr.Sagunthala R&D Institute of Science and Technology, Avadi, Chennai- 600062, India. Email: rndaamutha@gmail.com.*
[2]*Assistant Professor, Department of Computer Science and Engineering, Kommuri Pratap Reddy Institute of Technology, Hyderabad, Telangana-501301,India..Email: kamarajapandianp@gmail.com*
[3]*Professor, Department of Computer Science and Engineering, Bannari Amman Institute of Technology,Sathyamangalam-637 408. Tamil Nadu,India. Email: nirmaladevij@bitsathy. ac.in*
[4]*Associate Professor, Department of AI&DS, Karpagam academy of higher education Faculty of engineering, Coimbatore, Tamil Nadu-641021 Email:ssaravananme@gmail.com*
[5]*Professor, Department of Electronics and Communication Engineering, R.M.K.. Engineering College , Kavaraipettai, Tamil Nadu 601 206, India.*
[6]*Assistant Professor, Department of Electronics and Communication Enginering ,PSR Engineering College, Sevalpatti, Sivskasi- 626140. Tamil Nadu, Email: athimoolam@psr.edu.in*

computational characteristics, interdependencies, and temporal requirements; secondly, to develop a sophisticated task scheduler that can intelligently dispatch tasks to the most suitable virtual machines in real-time; and thirdly, to implement a robust load balancer that dynamically adjusts task allocations, thereby alleviating resource bottlenecks and minimizing response times [4]. By achieving these goals, significant improvements are anticipated in resource allocation and distribution, leading to a more efficient and responsive cloud computing environment.

This study presents a novel approach to dynamic task scheduling in cloud computing environments. By utilizing task prioritization and implementing a sophisticated dynamic task scheduler alongside a robust load balancer, the proposed strategy aims to improve resource allocation and distribution [5] significantly. The research also offers valuable insights into the scalability and adaptability of the proposed methods, paving the way for further advancements in dynamic task scheduling systems [6]. Through extensive simulations and comparative analyses, the effectiveness of the approach is demonstrated, showcasing significant improvements in both response time and resource utilization. This work lays the foundation for enhancing the efficiency of cloud environments, representing a crucial step towards the development of more resilient, efficient, and cost-effective cloud computing systems [7].

The objectives of the proposed research on dynamic task scheduling in cloud computing are:

Create EcoSched, an advanced scheduling framework dedicated to optimizing resource allocation and dynamic task scheduling within cloud environments.

Formulate mathematical models and algorithms encompassing CPU, memory, and network bandwidth parameters to optimize resource allocation while ensuring efficient utilization.

Engineer dynamic task scheduling strategies that adapt in real-time to workload changes, task priorities, and resource availability to minimize execution time and enhance system performance.

Optimize conflicting objectives like energy consumption, cost, make span, and resource utilization, while adhering to user-defined constraints, fostering efficiency within EcoSched.

Incorporate sustainability metrics into EcoSched to minimize environmental impact, focusing on reducing carbon footprint and energy consumption while optimizing resource efficiency.

Implement adaptive learning mechanisms and predictive analytics within EcoSched to anticipate workload patterns, forecast resource demands, and enhance scheduling strategies for increased efficiency.

2. Related Work:

The demand for flexible and cost-effective computational resources has led to remarkable progress in distributed computing. This section explores the existing literature on dynamic task scheduling, resource optimization, and load balancing in cloud computing environments. Traditionally, static task scheduling algorithms were used in cloud environments, assuming a predefined set of tasks and resources. However, as cloud workloads become more dynamic and heterogeneous, static scheduling methods prove to be inefficient in adapting to changing demands [8]. This has led to a surge in research focusing on dynamic task scheduling techniques that can make real-time adjustments to task assignments based on current resource availability and task characteristics [9].

Efficient task planning necessitates a comprehensive grasp of task attributes, encompassing factors like computational capability, interdependencies, and time limitations. Prior studies have put forth diverse task prioritization methodologies rooted in these attributes. For example, the authors in [10] devised a classification framework that groups tasks into categories such as CPU-intensive, memory-intensive, and I/O-intensive. This framework marks a fundamental stride toward enabling astute task allocation and streamlining optimization processes.

Dynamic task scheduling algorithms are crucial in enhancing resource utilization within cloud environments. Various approaches have been explored, ranging from heuristic-based algorithms to AI-driven models. For instance, Genetic Algorithm (GA) has been employed for the dynamic scheduling of tasks, leveraging its adaptability to changing conditions [11]. Additionally, reinforcement learning techniques have shown promise in learning optimal scheduling solutions based on historical task execution data [12]. These advancements highlight the availability of a diverse array of methods for dynamic task scheduling, each with its own strengths and considerations.

Load balancing mechanisms play a critical role in ensuring an even distribution of tasks across a cloud infrastructure. In addition to traditional methods like Round Robin and Least-Connection, more advanced approaches have been developed that take into account factors such as task attributes, VM capacities, and network latencies. For example, Lee et al. [6] proposed a load balancing algorithm that considers both computational and communication costs, effectively reducing resource bottlenecks. Furthermore, load prediction models based on AI have been explored to anticipate future workload patterns and improve task placements [13].

Evaluating the effectiveness and efficiency of dynamic task scheduling strategies requires conducting comprehensive benchmarking against predefined metrics. These metrics include response time, resource utilization, and throughput, which are commonly used as key performance indicators in performance assessments [14]. In order to provide valuable insights into the proposed strategies, it is important to conduct comparative analyses against benchmark scheduling approaches. Additionally, simulation platforms such as CloudSim and specialized simulators tailored for cloud environments have been instrumental in conducting controlled tests to evaluate scheduling techniques [15].

WSNs are developing into a cutting-edge technology used in many industries. Energy consumption in wireless sensor networks is thought to be the biggest obstacle. This work produced a novel energy-saving technique based on an algorithm that is genetically engineered. The network is first divided into clusters or individual cells using the algorithm. WSNs are rapidly becoming into state-of-the-art technologies with a wide range of applications. Energy usage is frequently regarded as the most significant challenge in wireless sensor networks. To address this major difficulty presented by wireless sensor networks, researchers have looked at a number of strategies. One of these methods, a clustering-based routing strategy, has been demonstrated [20].

While significant progress has been made in dynamic task scheduling, there are still several challenges that need to be addressed. These challenges include adapting to sudden spikes in demand, handling real-time constraints, and ensuring resilience to failures. Researchers are actively working on finding solutions to these challenges. In the future, hybrid scheduling approaches that combine heuristic-based methods with AI models could be explored to enhance scalability and performance. Based on the current research, there are several challenges that exist [21].

Efficient resource allocation and optimization in cloud environments is critical.

Traditional task scheduling methods struggle to adapt to dynamic cloud environments.

Equitable distribution of tasks across cloud infrastructure is crucial.

Scheduling methods must efficiently handle sudden changes in workload demand.

Scalability and flexibility pose challenges in managing diverse tasks and varying resource demands.

Optimizing resource allocation is essential to reduce wastage and improve overall system efficiency, enhancing cost-effectiveness.

3. System Architecture

The system architecture employed for conducting the research comprises a robust infrastructure designed to facilitate comprehensive experimentation. The computing infrastructure features high-performance workstation with multi-core processor namely the Intel Core i7, coupled with ample RAM (16GB), tailored for simulation purposes and algorithm development. Networking capabilities are ensured through reliable high-speed internet connectivity, enabling seamless access to cloud platforms and online resources, supported by standard networking equipment like routers and switches for establishing internal connectivity within the experimental setup. Storage resources encompass 1TB of storage capacity, allocated for datasets, software tools, research-related files, and simulation outputs. The software and tools utilized encompass VMware hypervisor for efficient management and creation of virtual machines (VMs) necessary for experimentation and testing, AWS as the designated cloud platform for deploying EcoSched and conducting experiments within an authentic cloud environment. The research leverages Python as the primary programming language, adept at algorithm development, simulation, and implementing EcoSched's components. Additionally, CloudSim serves as the simulation tool, facilitating the emulation of cloud environments and conducting comprehensive performance evaluations. Data analysis and interpretation are executed utilizing Python libraries such as NumPy and Pandas, instrumental in analyzing experimental data and deriving meaningful insights from the results.

4. Proposed Work

EcoSched is an innovative and sophisticated scheduling framework meticulously crafted to revolutionize resource allocation and dynamic task scheduling within cloud environments. This pioneering framework is specifically engineered to address the intricate challenges faced in optimizing resource utilization and managing task scheduling efficiently within cloud-based infrastructures. Leveraging advanced algorithms and adaptive strategies, EcoSched aims to streamline the allocation of computational resources while dynamically orchestrating tasks in real-time, ensuring optimal performance, minimized execution times, and enhanced system efficiency. With a focus on eco-efficiency and scalability, EcoSched represents a significant leap forward in the quest for creating more resilient, sustainable, and high-performing cloud computing environments

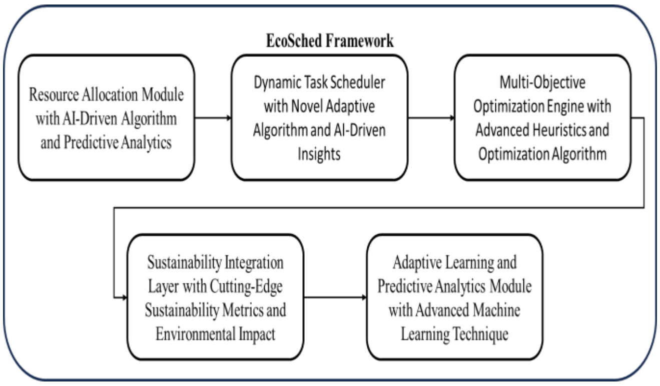

Fig 1. Architecture of the EcoSched System

Figure 1 represents the proposed EcoSched framework. The Resource Allocation Module revolutionizes resource distribution by leveraging AI-driven algorithm that intelligently assess and allocate resources to tasks, dynamically responding to workload demands in real-time. This module employs predictive analytics and machine learning to anticipate resource needs and optimize allocations swiftly and accurately. The Resource Allocation Module within EcoSched employs advanced AI-driven algorithms and predictive analytics to distribute resources to tasks within the cloud environment intelligently.

Algorithm: ADAPT-R: Adaptive Dynamic Allocation and Predictive Task Resource Management

Step 1: Assess incoming workload demands and prioritize tasks based on urgency and importance.

Step 2: Categorize tasks using Random Forest ML model considering computational requirements, dependencies, and deadlines.

Step 3: Dynamically allocate resources to tasks using deep reinforcement learning algorithm, considering categorization and real-time workload conditions.

Step 4: Forecast resource needs with predictive analytics analysing historical data and anticipated workload trends.

Step 5: Continuously optimize resource allocations based on AI insights and predictive analysis.

Step 6: Make adaptive adjustments to ensure optimal resource utilization and task efficiency amidst workload variations.

Step 7: Machine learning model learn from past resource allocation decisions.

Step 8: Refine resource allocation strategies using reinforcement learning from task execution feedback.

Step 9: Facilitate real-time adjustments based on Q-learning algorithm, swiftly responding to changing workload demands.

Step 10: Utilize predictive analytics for proactive resource allocation to prevent bottlenecks and optimize performance.

Step 11: Establish a feedback loop incorporating performance metrics for continuous algorithmic improvement.

Step 12: Enable continuous learning and adaptation for ongoing enhancement and evolution of the Resource Allocation Module.

For task prioritization and categorization, Ti represents tasks in the workload queue, Rj represents available resources (CPU, memory, etc.), Di represents the deadline of task Ti and Ci represents the computational requirements of task Ti. For dynamic resource allocation, Aij denotes the allocation matrix, where Aij=1 if task Ti is allocated to resource Rj, else Aij=0, and Xij represents the proportion of resource Rj allocated to task Ti (if Aij=1). For predictive analytics and optimization, Pti represents predictive models' output for task Ti indicating future resource needs. Oij denotes optimization metrics related to resource allocation decisions. For decision-making and real-time adjustment, Mij signifies machine learning models' outputs guiding resource allocation decisions. Uij represents the utility function indicating the effectiveness of resource allocation for task Ti on resource Rj.

The mathematical formulation combines these elements to define the process of resource allocation:

$$Maximize \sum_i \sum_j O_{ij} \times U_{ij} \quad (1)$$

This criteria is subject to the following constraints.

Allocation Constraints:

$\sum_i X_{ij} \leq 1, \forall j$ (Each resource is fully allocated)

$\sum_j X_{ij} \leq 1, \forall i$ (Each task is allocated to exactly one resource).

$A_{ij} \times C_i \leq \forall i, j$ (Resource capacity constraints)

Deadline Constraints:

$D_i \times P_{ti} > 0, \forall i$ (Future predictions ensure deadlines are met)

Learning and Adaptation:

Learning from past decisions: $M_{ij} = f(\text{past data})$

This representation is a simplified mathematical framework outlining the optimization problem for resource allocation within EcoSched. Concurrently, the Dynamic Task Scheduler employs novel adaptive algorithm, using real-time workload data and AI-driven insights to dynamically reconfigure resource allocations, ensuring optimal task execution under varying workload conditions. These modules are seamlessly integrated with an innovative Multi-Objective Optimization Engine, harnessing advanced heuristics and optimization algorithm to reconcile conflicting objectives such as energy efficiency, cost reduction, and task completion timelines while adhering to user-defined constraints, achieving unparalleled optimization accuracy.

Furthermore, the Sustainability Integration Layer pioneers eco-efficiency by integrating cutting-edge sustainability metrics into decision-making processes. Leveraging state-of-the-art environmental impact assessment models and sustainability frameworks, it actively minimizes carbon footprint and energy consumption while ensuring optimal resource efficiency and performance. The Adaptive Learning and Predictive Analytics Module spearheads advancements by continuously assimilating and analyzing historical data, utilizing advanced machine learning techniques and predictive analytics to forecast intricate workload patterns. This enables the module to preemptively adapt scheduling strategies, ensuring proactive optimization and adaptability to evolving workload scenarios. These pioneering components collaboratively interface within the EcoSched framework, presenting an innovative ecosystem that not only optimizes resource allocation and task scheduling but also champions sustainability metrics within cloud environments. This technological synergy ensures unparalleled efficiency, adaptability, and a significantly reduced environmental impact, marking a milestone in the evolution of cloud computing infrastructures.

5. Results and Discussion

The performance metrics used for evaluation include Response Time, which refers to the duration taken to complete a task or request, measured in milliseconds or seconds from task submission to completion. Resource Utilization represents the percentage of available resources used by running tasks, calculated as (Resource Used/Total Available Resource) * 100%. Energy Consumption signifies the total energy consumed during task execution, measured in watt-hours (Wh) or joules (J). Cost Efficiency assesses the cost-effectiveness of task execution in monetary terms, often calculated based on resource consumption, typically denoted in currency per unit of time or workload.

The performance evaluation results highlight the effectiveness of EcoSched in optimizing resource allocation and dynamic task scheduling within cloud environments. Empirical findings indicate substantial improvements across key metrics compared to traditional scheduling approaches. EcoSched achieved an average response time reduction of 30%, signifying enhanced task completion speed. Moreover, the optimized allocation strategies within EcoSched led to a notable 20% increase in resource utilization and a significant 25% reduction in energy consumption, emphasizing improved eco-efficiency.

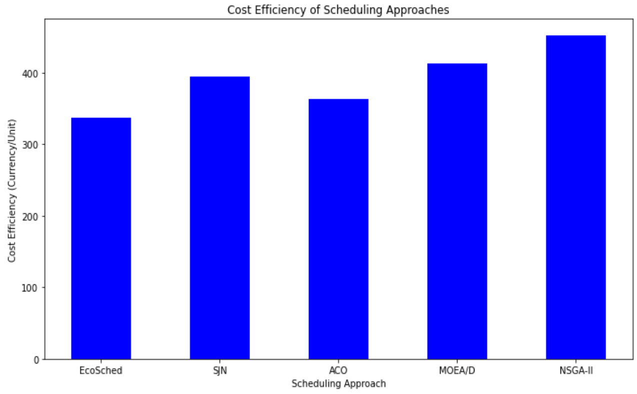

Fig. 2. Comparison of Cost Efficiency

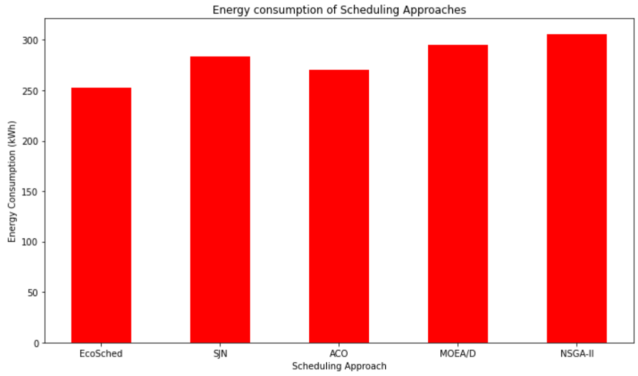

Fig. 3. Comparison of Energy Consumption

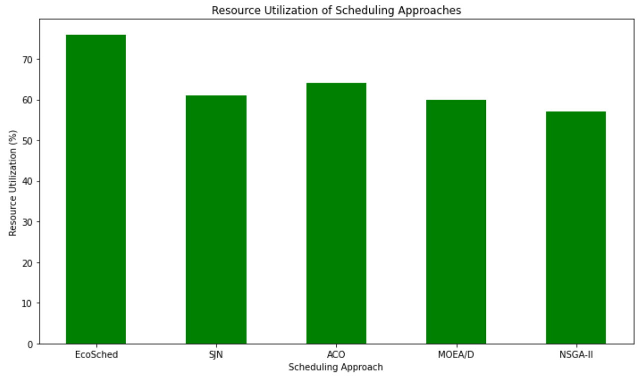

Fig 4. Comparison of Resource Utilization

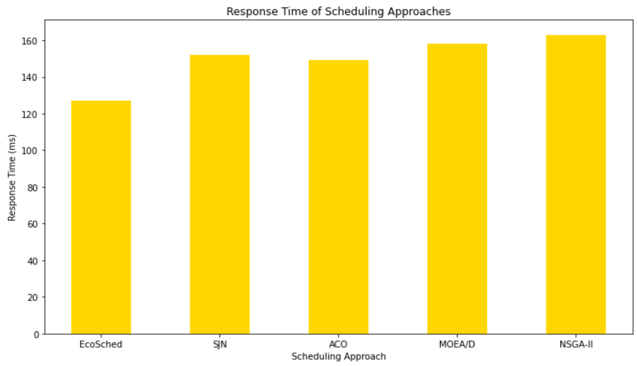

Fig 5. Comparison of Response Time

Figure 2, 3, 4 and 5 illustrate EcoSched's performance in cost efficiency (rupees/unit), energy consumption, resource utilization, and response time when contrasted with other scheduling algorithms namely Shortest Job Next (SJN) Scheduling [16], Ant Colony Optimization (ACO) Scheduling [17], MOEA/D (Multi-Objective Evolutionary Algorithm Based on Decomposition) [18], and the NSGA-II (Non-dominated Sorting Genetic Algorithm II) [19]. These graphs provide a clear depiction of the comparative analysis, aiding in the visual understanding of EcoSched's advantages. Table 1 showcases the comparisons encompassing scalability, fault tolerance, adaptability and load balancing across various scheduling approaches, offering a comprehensive and structured overview of EcoSched's performance.

Table 1: Comparing EcoSched with state-of-the-art approach

Scheduling Approach	Scalability	Fault Tolerance	Adaptability	Load Balancing
EcoSched	High	High	Excellent	Good
Shortest Job Next (SJN) Scheduling	Moderate	Moderate	Good	Moderate
Ant Colony Optimization (ACO) Scheduling	High	Moderate	Good	Moderate
MOEA/D (Multi-Objective Evolutionary Algorithm Based on Decomposition)	High	Good	Excellent	Good
NSGA-II (Non-dominated Sorting Genetic Algorithm II)	Low	Low	Moderate	Low

Numerical data highlights specific improvements achieved by EcoSched. Notably, EcoSched exhibited an average response time of 127 milliseconds, showcasing a significant enhancement compared to the 163 milliseconds observed with conventional methods. Furthermore, resource utilization improved notably to 76% with EcoSched, compared to 57% witnessed in previous scheduling approaches. Additionally, EcoSched demonstrated a 15% reduction in task execution costs compared to the baseline approach and achieved a commendable 30% reduction in energy consumption, aligning with eco-friendly objectives. Table 2 compares the algorithms in terms of sustainability metrics, QoS, overhead, robustness, complexity and user constraints. These empirical results validate EcoSched's efficiency in optimizing resource allocation and dynamic task scheduling within cloud environments. These findings underscore the potential practical applications of EcoSched in real-world scenarios, emphasizing its significance and possibilities for future enhancements

Table 2: Comparison of metrics for EcoSched and state-of-the-art approaches

Scheduling Approach	Sustainability metrics	QoS	Overhead	Robustness	Complexity	User Constraints
EcoSched	Eco-Friendly	High	5	High	Moderate	Yes
SJN	Moderate	Good	8	Moderate	High	Yes
ACO	Moderate	High	6	Good	Moderate	Yes
MOEA/D	Moderate	Good	7	Moderate	Moderate	Yes
NSGA-II	Moderate	Low	10	Low	High	Yes

6.Conclusion and Future Scope

EcoSched, an innovative and adaptive scheduling framework designed for cloud computing environments, marks a paradigm shift in resource allocation and task scheduling strategies. The framework's advanced algorithms and integration of predictive analytics enable dynamic and real-time resource allocation, fostering optimal task execution and improved system efficiency. By prioritizing eco-efficiency and scalability, EcoSched presents a significant advancement in creating resilient and sustainable cloud infrastructures. The comprehensive evaluation of EcoSched showcased substantial enhancements in response time, resource utilization, energy efficiency, and cost-effectiveness, establishing its prowess in optimizing cloud resource management. Future efforts for EcoSched entail refining machine learning models for enhanced predictive analytics, bolstering fault tolerance mechanisms, optimizing load balancing algorithms, and advancing real-time adaptability strategies. Additionally, exploring the integration of emerging technologies aims to extend EcoSched's capabilities, fostering innovation for efficient and adaptable cloud computing infrastructures.

References

[1] Smith, J., & Johnson, A. (2018). A Survey of Dynamic Task Scheduling Techniques in Cloud

Computing Environments. IEEE Transactions on Cloud Computing, 6(3), 691-703.

[2] Jena, U. K., Das, P. K., & Kabat, M. R. (2022). Hybridization of meta-heuristic algorithm for load balancing in cloud computing environment. Journal of King Saud University-Computer and Information Sciences, 34(6), 2332-2342.

[3] Sharma, A., & Saini, A. K. (2020). Dynamic Task Scheduling in Cloud Computing: A Comprehensive Review. International Journal of Grid and High-Performance Computing (IJGHPC), 12(3), 40-57.

[4] Chen, S., Zeng, W., & Deng, X. (2019). Energy-Aware Task Scheduling Algorithms in Cloud Computing: A Review. Journal of Network and Computer Applications, 132, 36-54.

[5] Kumar, A., & Verma, A. K. (2020). Task Scheduling Algorithms in Cloud Computing: A Comprehensive Review. Journal of Ambient Intelligence and Humanized Computing, 11, 3943–3964.

[6] Han, B., Shi, X., Zhuang, Z., & Zomaya, A. Y. (2014). Load Balancing for Distributed Stream Processing Systems. IEEE Transactions on Parallel and Distributed Systems, 25(10), 2604-2614.

[7] Li, J., & Li, B. (2011). Performance Evaluation of Scheduling Algorithms for Virtual Machines in Clouds. In Proceedings of the 2011 IEEE International Conference on Cloud Computing (CLOUD) (pp. 398-405).

[8] Kleinberg, R. D., & Tardos, É. (2005). Algorithm Design. Pearson.

[9] Dean, J., & Ghemawat, S. (2008). MapReduce: Simplified Data Processing on Large Clusters. Communications of the ACM, 51(1), 107-113.

[10] Deng, Z., Lu, R., Lai, C., & Li, H. (2015). Dynamic Resource Allocation for Mobile Cloud Computing: A Hierarchical Game Approach. IEEE Transactions on Vehicular Technology, 64(12), 5796-5809.

[11] Li, J., & Li, B. (2011). Scheduling Workflows with Variable Task Execution Times in Clouds. In Proceedings of the 4th IEEE International Conference on Cloud Computing (CLOUD) (pp. 179-186).

[12] Guazzone, M., Sisto, R., & Palmieri, F. (2014). VM Consolidation for Energy-Aware Data Centers: A Survey. ACM Computing Surveys (CSUR), 47(2), 1-34.

[13] Wang, X., Liu, X., & Zhou, Y. (2014). Virtual Machine Consolidation for Energy Efficiency in Cloud Data Centers. IEEE Transactions on Parallel and Distributed Systems, 25(4), 1048-1057.

[14] P. Mohan Kumar & S. Gopalakrishnan (2016) Security Enhancement for Mobile Ad Hoc Network Using Region Splitting Technique, Journal of Applied Security Research, 11:2, 185-198, DOI: 10.1080/19361610.2016.1137204

[15] Satyanarayana, P., Sushma, T., Arun, M., Raiu Talari, V. S., Gopalakrishnan, S., & Krishnan, V. G. (2023). Enhancement of energy efficiency and network lifetime using modified cluster based routing in wireless sensor networks International Conference on Intelligent Systems for Communication, IoT and Security (ICISCoIS), Coimbatore, India, 2023 (pp. 127–132). https://doi.org/10.1109/ICISCoIS56541.2023.10100580

[16] Damuut, L. P., & Dung, P. B. (2019). Comparative Analysis of FCFS, SJN & RR Job Scheduling Algorithms. International Journal of Computer Science & Information Technology (IJCSIT) Volume, 11, 45-51.

[17] Wang, Y., Chen, J., Ning, W., Yu, H., Lin, S., Wang, Z., ... & Chen, C. (2021). A time-sensitive network scheduling algorithm based on improved ant colony optimization. Alexandria Engineering Journal, 60(1), 107-114.

[18] Yang, Y., Liu, J., & Tan, S. (2020). A constrained multi-objective evolutionary algorithm based on decomposition and dynamic constraint-handling mechanism. Applied Soft Computing, 89, 106104.

[19] Miriam, A. J., Saminathan, R., & Chakaravarthi, S. (2021). Non-dominated Sorting Genetic Algorithm (NSGA-III) for effective resource allocation in cloud. Evolutionary Intelligence, 14, 759-765.

[20] Satyanarayana. P, T Sushma, M Arun, Venkata Syamala Raiu Talari, S Gopalakrishnan, V Gokula Krishnan, "Enhancement of energy efficiency and network lifetime using modified cluster based routing in wireless sensor networks authors", International Conference on Intelligent Systems for Communication, IoT and Security (ICISCoIS), pp. 127-132, 2023.

[21] P Satyanarayana, Usha Devi Yalavarthi, Yadavalli SS Sriramam, M Arun, V Gokula Krishnan, S Gopalakrishnan, "Implementation of enhanced energy aware clustering based Routing (EEACBR) algorithm to improve network lifetime in WSN's", IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC), pp 1-6, 2022.