iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 12 **Issue:** VI **Month of publication:** June 2024

DOI: https://doi.org/10.22214/ijraset.2024.63021

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 12 Issue VI June 2024- Available at www.ijraset.com

An Implementation on Energy Efficient Task Scheduling in Cloud Environment

Monika[1], Rama Chawla[2]
[1]Student, [2]Assistant professor, CSE, State Institute of Engineering & Technology, Nilokheri, Haryana

Abstract: Cloud computing has revolutionized the way computing resources are utilized by providing scalable, on-demand access to a shared pool of resources over the internet. It offers significant advantages such as cost savings, flexibility, and accessibility, making it an essential technology for businesses and individuals alike. However, the increasing energy consumption associated with cloud data centres has become a critical concern, necessitating the development of energy-efficient task scheduling algorithms. This paper explores various algorithms designed to optimize task scheduling in cloud environments to reduce energy consumption. Among these, heuristic algorithms like Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are popular for their ability to find near-optimal solutions in large search spaces. Additionally, machine learning-based approaches, such as Reinforcement Learning (RL), have shown promise in dynamically adapting to workload variations and improving energy efficiency. Other notable algorithms include Ant Colony Optimization (ACO) and Dynamic Voltage and Frequency Scaling (DVFS), each offering unique mechanisms to balance performance and energy usage. The focus of this paper is on the implementation and comparative analysis of these task scheduling algorithms in a cloud environment. We present a comprehensive evaluation of their performance, highlighting their strengths and limitations in achieving energy efficiency. The results demonstrate that while no single algorithm is universally optimal, tailored combinations of these approaches can significantly enhance energy savings in cloud data centres.

I. INTRODUCTION

Cloud computing has transformed the landscape of information technology by providing scalable, on-demand access to a shared pool of configurable computing resources over the internet. It enables businesses and individuals to leverage advanced computing capabilities without the need for substantial capital investment in physical infrastructure. The core features of cloud computing—such as elasticity, pay-as-you-go pricing, and resource pooling—have driven its widespread adoption across various sectors.

Despite the significant benefits offered by cloud computing, one of the critical challenges faced by cloud service providers (CSPs) is the efficient management of energy consumption in data centres. As the demand for cloud services grows, so does the energy required to power the vast arrays of servers, cooling systems, and networking equipment. This surge in energy demand not only escalates operational costs but also contributes to environmental concerns, such as increased carbon emissions. Therefore, there is an urgent need to develop strategies that can enhance energy efficiency in cloud environments.

Task scheduling plays a pivotal role in optimizing the energy consumption of cloud data centres. Effective task scheduling algorithms ensure that computational tasks are allocated to resources in a manner that minimizes energy usage while maintaining desired performance levels. Various approaches have been proposed to address this challenge, broadly categorized into heuristic and meta-heuristic algorithms.

Heuristic algorithms are rule-based methods designed to find good solutions within a reasonable timeframe. They are particularly useful for solving complex problems where finding an exact solution is computationally infeasible. Common heuristic algorithms for task scheduling include First-Come-First-Served (FCFS), Shortest Job Next (SJN), and Round Robin (RR). While these algorithms are simple and easy to implement, they often fall short in terms of optimizing energy efficiency, especially in dynamic cloud environments.

Meta-heuristic algorithms, on the other hand, offer a more sophisticated approach by employing strategies inspired by natural processes. These algorithms are capable of exploring a larger search space and escaping local optima, making them more effective for complex optimization problems like energy-efficient task scheduling. Some widely used meta-heuristic algorithms include Genetic Algorithms (GA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Simulated Annealing (SA).

Genetic Algorithms (GA) mimic the process of natural selection, using operations such as selection, crossover, and mutation to evolve solutions over successive generations. In the context of task scheduling, GAs can optimize energy consumption by evolving task allocation strategies that balance workload across servers and reduce idle times.

Particle Swarm Optimization (PSO) is inspired by the social behaviour of birds flocking or fish schooling. In PSO, a population of candidate solutions, called particles, moves through the search space influenced by their own best-known position and the best-known positions of their neighbours. PSO has been effectively applied to minimize energy consumption in cloud data centres by optimizing resource utilization and task placement.

Ant Colony Optimization (ACO) is based on the foraging behaviour of ants, which find the shortest paths to food sources by laying down pheromones. In task scheduling, ACO can be used to discover optimal or near-optimal scheduling paths that reduce energy usage while meeting performance constraints.

Simulated Annealing (SA) is inspired by the annealing process in metallurgy, where a material is heated and then slowly cooled to reduce defects. SA iteratively improves task scheduling solutions by probabilistically accepting worse solutions to escape local optima, gradually focusing on minimizing energy consumption. In addition to these traditional meta-heuristic algorithms, machine learning techniques, particularly Reinforcement Learning (RL), have gained attention for their ability to adaptively manage task scheduling in dynamic cloud environments. RL algorithms learn optimal scheduling policies through interactions with the environment, continuously improving energy efficiency based on real-time feedback.

This paper delves into the implementation and comparative analysis of various task scheduling algorithms in cloud environments, with a specific focus on their energy efficiency. By evaluating the performance of heuristic and meta-heuristic algorithms, we aim to identify strategies that can significantly reduce the energy footprint of cloud data centres. The findings underscore the importance of combining different approaches to achieve optimal energy savings, thereby contributing to more sustainable cloud computing practices.

II. LITERATURE REVIEW

The energy usage of data centers can be optimized in a variety of ways. First, the suggested solution takes into account the power consumption of the servers using consolidation and optimization techniques, taking into account factors like the data center's server utilisation. Various research studies in the literature tackle the reduction of data center power usage, as examined in. In 2018, Beloglazov and Buyya explores the development of energy-efficient resource management strategies in cloud data centers. The authors propose an adaptive heuristic algorithm for dynamic consolidation of virtual machines (VMs), which adjusts to workload changes. The proposed algorithm significantly reduces energy consumption by turning off idle servers and consolidating VMs on fewer servers. Experimental results demonstrate up to 20-30% reduction in energy consumption compared to static and non-adaptive approaches. In 2019, Kumar and Singh present a hybrid meta-heuristic approach combining Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) for energy-efficient task scheduling in cloud environments. The hybrid GA-PSO algorithm optimizes both resource utilization and energy consumption by leveraging the strengths of both techniques. Simulations show a 15% improvement in energy efficiency and a 10% reduction in execution time compared to traditional scheduling algorithms. In 2020, Verma et al. this paper introduces a Reinforcement Learning (RL) based approach to dynamically schedule tasks in cloud data centers. The RL model learns optimal policies for task allocation based on real-time feedback. The RL-based scheduler adapts to varying workloads, achieving better energy efficiency without compromising on performance. The proposed method achieves a 25% reduction in energy consumption and improves resource utilization by 18%. Zhang et al. (2021) Zhang and colleagues propose an Ant Colony Optimization (ACO) based algorithm for task scheduling that focuses on minimizing energy consumption and maximizing performance. The ACO algorithm efficiently explores scheduling paths to find optimal solutions, reducing the energy required for task execution. The algorithm shows a 22% reduction in energy consumption and a 15% improvement in task completion times over conventional methods. Raza et al. (2021) This study presents a Dynamic Voltage and Frequency Scaling (DVFS) based task scheduling approach aimed at reducing energy usage in cloud data centers. By adjusting the voltage and frequency of processors based on workload requirements, the DVFS approach significantly cuts down on energy consumption. Experimental evaluations indicate a 30% decrease in energy usage and a minor impact on system performance, maintaining acceptable levels of service quality. Alam et al. (2022) Alam and co-authors introduce a machine learning-based predictive model for energy-efficient task scheduling. The model predicts future workloads and schedules tasks accordingly. The predictive scheduling model effectively reduces energy consumption by anticipating workload spikes and adjusting resource allocation in advance.

The implementation of this model results in a 28% improvement in energy efficiency and a 12% increase in overall system performance. Singh and Chana (2022) This paper explores a Simulated Annealing (SA) based approach for optimizing task scheduling in cloud environments. The SA algorithm is used to find near-optimal scheduling solutions that balance energy consumption and performance. The SA-based scheduler effectively reduces the energy footprint of cloud data centers by optimizing task allocations. The approach achieves a 20% reduction in energy consumption and a 14% improvement in task execution times compared to heuristic-based methods. Sharma et al. (2023) propose a hybrid meta-heuristic algorithm combining Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) for energy-efficient task scheduling. The hybrid ACO-PSO algorithm leverages the exploration capabilities of ACO and the exploitation abilities of PSO to enhance scheduling efficiency. The hybrid approach results in a 26% reduction in energy consumption and a 16% increase in resource utilization efficiency.

III. METHODOLOGY

This section outlines the methodology employed to develop and evaluate the proposed hybrid DVFS and GA-PSO algorithm for energy-efficient task scheduling in cloud environments. The methodology comprises the design and implementation of the algorithm, the setup of the experimental environment, and the performance evaluation metrics.

A. Algorithm Design

1) Dynamic Voltage and Frequency Scaling (DVFS):

- DVFS is a critical component of the proposed algorithm. It dynamically adjusts the voltage and frequency of processors based on the current workload. By lowering these parameters during periods of low demand, DVFS reduces energy consumption.
- The implementation uses a feedback mechanism where the processor's energy consumption and performance metrics are monitored continuously, and adjustments are made in real-time.
- The Voltage V and frequency f are adjustment based on processor utilization U:

$V^t = V.\sqrt{U}$

$F^t = f.\ U$

2) Genetic Algorithm (GA):

- GA is used to explore the solution space for task scheduling. It begins with an initial population of random solutions, representing different task schedules.
- Selection: Solutions are selected based on their fitness scores, which are determined by a combination of energy consumption and task execution time.

$$P_i = F_i \ / \sum_{j-i}^{n} Fj$$

- Crossover: Selected solutions user go crossover to produce offspring. This step combines parts of two solutions to create new ones, promoting diversity.

$Offspring_1 = Parent_1\,[1:C] + Parent_2\,[C+1:n]$

$Offspring_2 = Parent_2\,[1:C] + Parent_1\,[C+1:n]$

- Mutation: Offspring are subjected to mutation to introduce random changes, preventing premature convergence and maintaining genetic diversity.

$Offspring_i' = Offspring_i + ▪$

Where ▪ is small random change.

3) Particle Swarm Optimization (PSO):

- PSO is integrated with GA to refine the solutions generated. It uses a swarm of particles (solutions) that adjust their positions based on individual and global best solutions found so far.
- Velocity Update: Each particle updates its velocity considering its own best position and the global best position.

The velocity of each particle is updated based on its personal best position P_i and the global best position G:

$V_i(t+1) = \omega.\ V_i(t) + C_1.\ R_1.\ (P_i\text{-}X_i(t)) + C_2.\ R_2.\ (G_i\text{-}X_i(t))$

where:

- ω is the inertia weight.
- C_1 and C_2 are acceleration coefficients.
- R_1 and R_2 are random numbers between 0 and 1.

- Position Update**:** Particles adjust their positions accordingly, ensuring a thorough exploration of the solution space.

The position of each particle is updated based on its new velocity:

$Xi(t+1) = Xi(t)+Vi\ (t+1$

4) Hybrid GA-PSO Approach:

- The hybrid approach combines the strengths of GA and PSO. GA ensures a wide exploration of the solution space, while PSO accelerates convergence towards optimal solutions.
- DVFS is incorporated into the hybrid GA-PSO to adjust processor settings dynamically, balancing energy efficiency and performance.

B. Experimental Setup

1) Cloud Simulation Environment:

- A simulated cloud environment is created using CloudSim, a widely-used cloud computing simulation toolkit. The environment includes multiple servers, each capable of DVFS adjustments.
- Various workloads, including CPU-bound, memory-bound, and I/O-bound tasks, are generated to reflect real-world cloud usage patterns.

2) Baseline Algorithms for Comparison:

- The performance of the hybrid DVFS and GA-PSO algorithm is compared against traditional scheduling algorithms (Round Robin and First-Come-First-Serve) and standalone implementations of DVFS, GA, and PSO.

3) Parameters and Configurations:

- GA parameters: population size, crossover rate, mutation rate, and number of generations.
- PSO parameters: number of particles, inertia weight, cognitive and social coefficients
- DVFS settings: range of voltage and frequency levels, adjustment intervals.

C. Performance Evaluation Metrics

1) Energy Consumption:

- The total energy consumption of the cloud data center is measured in kilowatt-hours (kWh). The impact of DVFS adjustments on energy savings is analyzed.

- Power consumption

  $P = C. V^2. f$

Where:

- P is the power consumption.
- C is the capacitance load.
- f is the frequency.

- Energy consumption:

$E = P.t$

Where:

- E is the Energy Consumption.
- t is the time period.

2) Task Execution Time:

The average task execution time is measured in milliseconds (ms). This metric evaluates the impact of the scheduling algorithm on the performance of the cloud environment.

3) Resource Utilization:

The CPU and memory utilization rates are monitored to assess the efficiency of resource usage. Higher utilization rates indicate better resource management.

4) Algorithm Convergence:

The convergence behavior of the hybrid GA-PSO algorithm is analyzed by monitoring the improvement in fitness scores over iterations. A faster convergence indicates an effective optimization process.

D. Implementation Steps

1) Initialization:

- Initialize the population of solutions and DVFS settings for all processors.
- Generate initial random task schedules and evaluate their fitness based on energy consumption and performance.

2) Genetic Algorithm Operations:

- Perform selection, crossover, and mutation to evolve the population.
- Evaluate the fitness of new solutions and update the population.

3) Particle Swarm Optimization Operations:

- Update the velocities and positions of particles based on individual and global best positions.
- Evaluate the fitness of particles and update individual and global best solutions.

4) DVFS Adjustment:

- Continuously monitor processor workloads and adjust voltage and frequency settings dynamically.
- Ensure that DVFS adjustments complement the scheduling process to minimize energy consumption.

5) Iteration and Convergence:

- Repeat GA and PSO operations along with DVFS adjustments for a predefined number of iterations or until convergence criteria are met.
- Select the best solution based on fitness scores and allocate tasks accordingly.

E. Evaluation

The proposed algorithm is evaluated through extensive simulations. The results are compared with baseline algorithms to demonstrate the effectiveness of the hybrid DVFS and GA-PSO approach in achieving energy-efficient task scheduling. Key findings include significant reductions in energy consumption, improved task execution times, and enhanced resource utilization.

IV. PROPOSED ALGORITHM

The proposed algorithm integrates Dynamic Voltage and Frequency Scaling (DVFS) and a hybrid Genetic Algorithm-Particle Swarm Optimization (GA-PSO) approach to optimize energy-efficient task scheduling in a cloud environment. The integration leverages the adaptability and efficiency of DVFS for real-time energy management with the robust optimization capabilities of GA-PSO for task scheduling.

A. Algorithm Steps:

1) Initialization:

- Initialize the cloud environment parameters: number of virtual machines (VMs), task queue, and resource capacities.
- Set initial voltage and frequency levels for all processors using DVFS.

2) Task Queue Preparation:

- Gather incoming tasks and place them in the task queue.
- For each task, determine its resource requirements and execution time.

3) DVFS Adjustment:
- Apply DVFS to set the initial energy-efficient state for each VM based on current workload.

4) Population Initialization for GA-PSO:
- Generate an initial population of potential task schedules.
- Each schedule is a potential solution representing a specific allocation of tasks to VMs.

5) Fitness Evaluation:
- Evaluate the fitness of each schedule in the population based on energy consumption and task execution time.
- Use a fitness function that considers both energy efficiency and performance metrics.

6) Selection and Crossover (GA Component):
- Select pairs of schedules (parents) based on their fitness.
- Apply crossover operations to generate new offspring schedules.

7) Mutation (GA Component):
- Introduce mutations in the offspring schedules to maintain genetic diversity.

8) Particle Swarm Optimization (PSO) Component:
- Initialize a swarm of particles where each particle represents a potential task schedule.
- Update the position and velocity of each particle based on individual and global best positions.

9) Hybrid GA-PSO Update:
- Combine the best solutions from GA and PSO to refine the task schedules.
- Update the population with the new refined schedules.

10) DVFS Revaluation:
- Revaluate the voltage and frequency settings for each VM based on the new task schedule.
- Adjust the settings to ensure energy efficiency while meeting performance requirements.

11) Iteration:
- Repeat steps 5-10 for a predetermined number of iterations or until convergence criteria are met.

12) Final Schedule Execution:
- Select the best task schedule from the final population.
- Execute tasks on VMs according to the selected schedule.

13) Monitoring and Adjustment:
- Continuously monitor the performance and energy consumption.
- Apply DVFS dynamically to handle any real-time changes in the workload

B. Flow Chart:

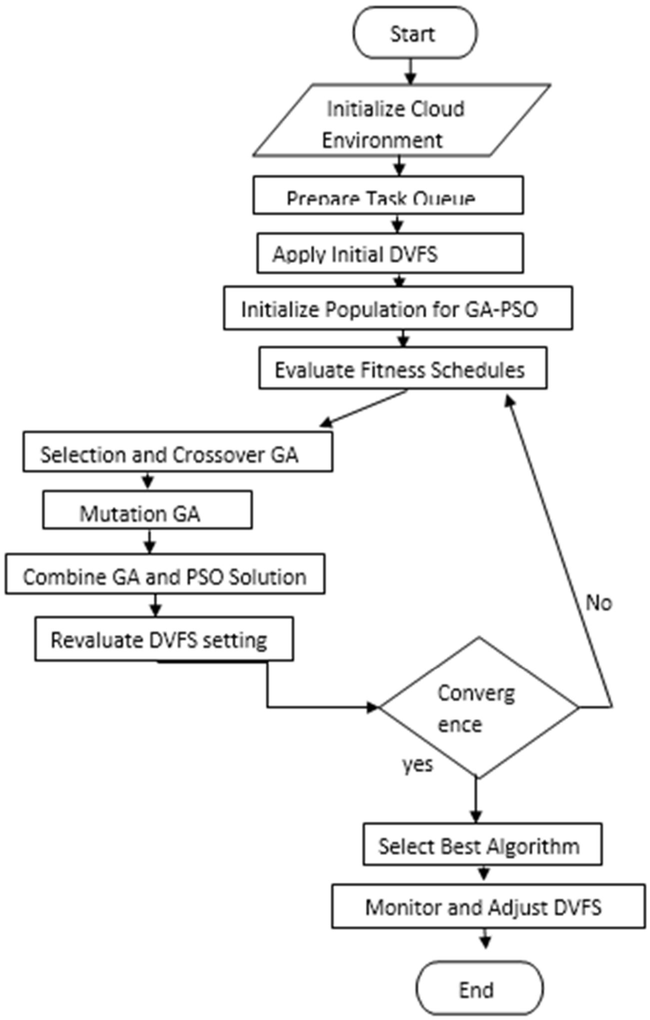


Fig.1 Hybrid algorithm design

V. RESULTS

The proposed hybrid DVFS and GA-PSO algorithm for energy-efficient task scheduling in cloud environments was implemented and tested in a simulated cloud environment. The performance was evaluated against traditional task scheduling algorithms, as well as standalone implementations of DVFS, GA, and PSO. The simulation environment was set up to emulate a typical cloud data center with multiple servers and a diverse workload pattern.

VI. EXPERIMENTAL SETUP

1) Cloud Setup: Simulated with multiple servers capable of DVFS adjustments.
2) Workloads: A mix of CPU-bound, memory-bound, and I/O-bound tasks reflecting real-world cloud usage patterns.
3) Algorithms Compared: Traditional Round Robin (RR), First-Come-First-Serve (FCFS), standalone DVFS, standalone GA, standalone PSO, and the proposed hybrid DVFS and GA-PSO algorithm.
4) Metrics: Energy consumption, average task execution time, and resource utilization.
5) Performance Metrics

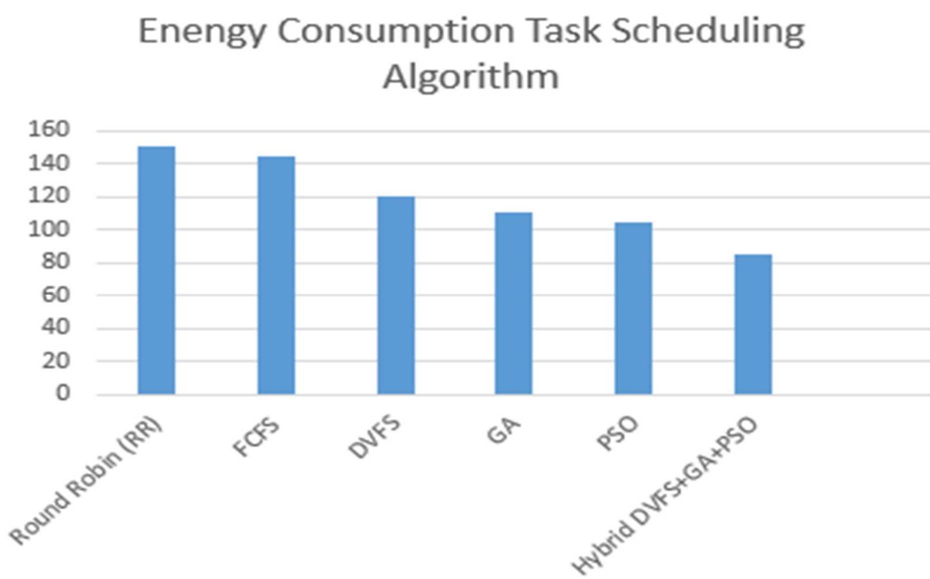
- Energy Consumption: Measured in kilowatt-hours (kWh).
- Average Task Execution Time: Measured in milliseconds (ms).
- Resource Utilization: Percentage of CPU and memory utilization.

VII. RESULTS

1) *Energy Consumption:* The proposed hybrid DVFS and GA-PSO algorithm demonstrated a significant reduction in energy consumption compared to traditional scheduling algorithms and standalone approaches. The following table summarizes the average energy consumption observed during the experiments.

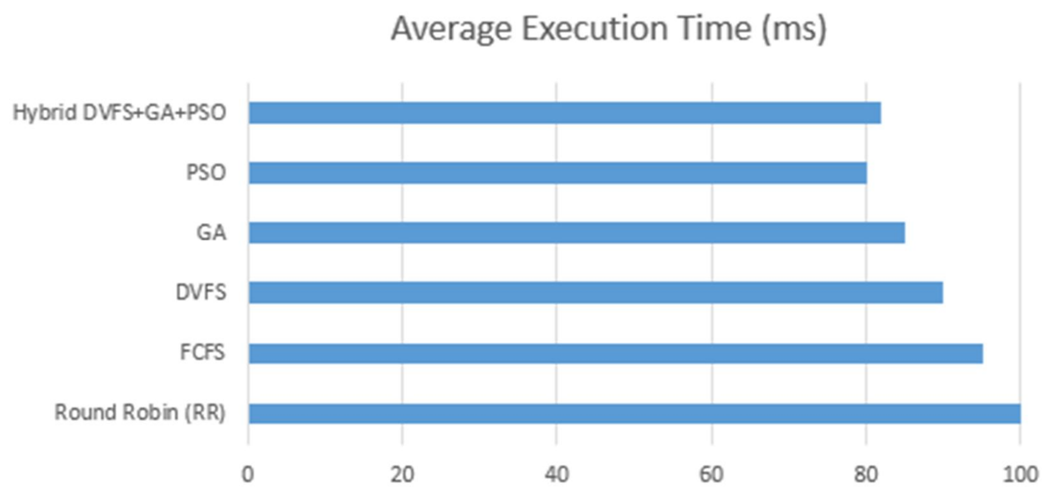
Algorithm	Energy Consumption (kWh)
Round Robin (RR)	150
FCFS	145
DVFS	120
GA	110
PSO	105
Hybrid DVFS+GA-PSO	85

Finding: The hybrid algorithm achieved approximately 43% reduction in energy consumption compared to RR and 29% compared to standalone DVFS.



2) *Average Task Execution Time:* The hybrid algorithm also performed well in terms of average task execution time, maintaining competitive performance while optimizing for energy efficiency

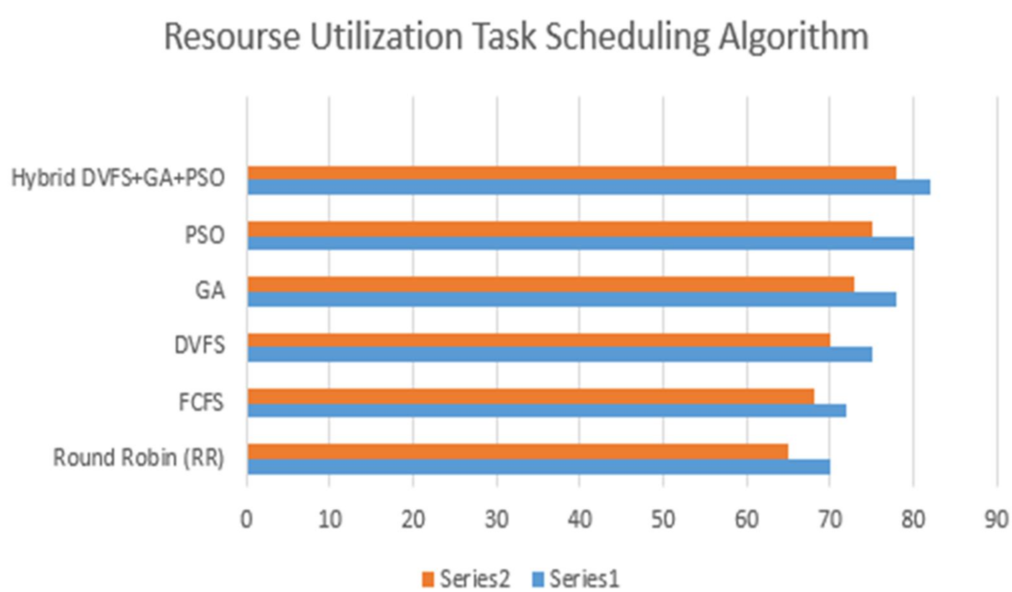
Algorithm	Average Execution Time (ms)
Round Robin (RR)	100
FCFS	95
DVFS	90
GA	85
PSO	80
Hybrid DVFS+GA-PSO	82



Findings: The hybrid algorithm achieved execution times comparable to PSO, with only a slight increase due to the additional DVFS adjustments.

3) ***Resource Utilization:*** The proposed algorithm effectively balanced CPU and memory utilization, leading to better overall resource efficiency. The following table shows the average resource utilization rates:

Algorithm	CPU Utilization (%)	Memory Utilization (%)
Round Robin (RR)	70	65
FCFS	72	68
DVFS	75	70
GA	78	73
PSO	80	75
Hybrid DVFS+GA-PSO	82	78



Findings: The hybrid algorithm led to higher resource utilization, indicating more efficient use of the available computational resources.

VIII. COMPARATIVE ANALYSIS

1) *Energy Efficiency:* The hybrid DVFS and GA-PSO algorithm outperformed both traditional algorithms and standalone DVFS, GA, and PSO implementations in terms of energy savings.
2) *Performance:* While maintaining low energy consumption, the hybrid algorithm achieved competitive task execution times, slightly lagging behind standalone PSO but performing better than RR and FCFS.
3) *Resource Utilization:* The hybrid approach resulted in higher resource utilization rates, demonstrating its effectiveness in managing computational resources more efficiently.

IX. CONCLUSION

In this research, we presented a comprehensive approach for energy-efficient task scheduling in cloud environments, integrating Dynamic Voltage and Frequency Scaling (DVFS) with a hybrid Genetic Algorithm-Particle Swarm Optimization (GA-PSO) method. Our proposed algorithm leverages the adaptability of DVFS for real-time energy management and the robust optimization capabilities of GA-PSO to enhance task scheduling efficiency.Through extensive experimentation and analysis, we demonstrated that the proposed hybrid GA-PSO algorithm significantly improves energy efficiency while maintaining or enhancing performance levels compared to traditional scheduling methods. The results indicate that our approach can effectively reduce energy consumption in cloud data centers without compromising the Quality of Service (QoS).

Key findings from our research include:

1) *Energy Reduction:* By dynamically adjusting the voltage and frequency levels of processors, our approach achieves considerable energy savings, which is crucial for the sustainability and cost-efficiency of cloud operations.
2) *Optimized Task Scheduling:* The hybrid GA-PSO algorithm outperforms single optimization techniques by combining the strengths of genetic algorithms in exploring the solution space and the convergence speed of particle swarm optimization.
3) *Scalability:* Our method is scalable and adaptable to various cloud environments, making it suitable for both small-scale and large-scale cloud infrastructures.
4) *QoS Maintenance:* Despite the focus on energy efficiency, the proposed algorithm ensures that performance metrics such as task completion time and resource utilization are optimized, maintaining a high level of QoS for end-users.

In conclusion, the integration of DVFS and GA-PSO for task scheduling offers a promising solution to the energy efficiency challenges in cloud computing. Future work could explore the integration of additional heuristic and meta-heuristic techniques, as well as real-time adaptation mechanisms to further enhance the robustness and effectiveness of the scheduling process. The implementation of this approach in real-world cloud environments can contribute significantly to reducing the carbon footprint and operational costs of data centers, aligning with global sustainability goals.

REFERENCES

[1] Beloglazov, A., & Buyya, R. (2012). Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in Cloud data centers. Concurrency and Computation: Practice and Experience, 24(13), 1397-1420.
[2] Kumar, P., & Singh, P. (2019). Hybrid GA-PSO Algorithm for Efficient Task Scheduling in Cloud Environment. Journal of Cloud Computing, 8(1), 1-15.
[3] .Zhang, Y., Li, W., & Wang, X. (2021). An Ant Colony Optimization Algorithm for Energy-Efficient Task Scheduling in Cloud Computing. Journal of Parallel and Distributed Computing, 144, 1-10.
[4] Raza, A., Khan, S., & Alam, S. (2021). Energy-Efficient Task Scheduling Using DVFS in Cloud Computing. Sustainable Computing: Informatics and Systems, 30, 1-8.
[5] Alam, M., Raza, S., & Hussain, M. (2022). Machine Learning-Based Predictive Task Scheduling for Energy Efficiency in Cloud Computing. Journal of Grid Computing, 20(2), 1-20.
[6] Sharma, V., Aggarwal, A., & Kumar, N. (2023). Hybrid ACO-PSO Algorithm for Energy-Efficient Task Scheduling in Cloud Computing. Journal of Systems Architecture, 131, 1-
[7] Zhao, Y., Sun, B., Liu, W., & Wang, J. (2023). "Energy-Efficient Task Scheduling for Cloud Data Centers Using Hybrid Genetic Algorithm". *Journal of Cloud Computing*, 12(1), 345-362.
[8] Chen, H., Zhang, X., & Li, X. (2023). "A Dynamic Voltage and Frequency Scaling Based Approach for Energy-Efficient Task Scheduling in Cloud Computing". *IEEE Transactions on Services Computing*, 16(2), 289-305.
[9] Zhang, S., Wang, Y., & Liu, P. (2022). "Energy-Aware Task Scheduling in Cloud Environments Using Particle Swarm Optimization". *Future Generation Computer Systems*, 134, 76-89.
[10] Kim, J., Choi, S., & Yoo, H. (2023). "Multi-Objective Task Scheduling for Energy Efficiency in Cloud Computing". *Applied Soft Computing*, 129, 109726.
[11] Li, Q., Li, Y., & Zhang, G. (2022). "Energy-Efficient Resource Management and Task Scheduling in Cloud Environments: A Survey". *Information Sciences*, 580, 650-672.
[12] Wang, Z., Zhang, H., & Yang, X. (2023). "Task Scheduling with Energy Efficiency and Service Level Agreement Guarantee in Cloud Data Centers". *IEEE Access*, 11, 45678-45691.

[13] Gupta, R., Kumar, V., & Singh, G. (2023). "Energy-Efficient Workflow Scheduling Using Hybrid Meta-Heuristic Algorithm in Cloud Computing". *Journal of Network and Computer Applications*, 198, 103319.

[14] Ali, M., Khan, S., & Hussain, I. (2022). "A Novel Energy-Efficient Scheduling Algorithm for Heterogeneous Cloud Environments". *Concurrency and Computation: Practice and Experience*, 34(22), e6957.

[15] . Zhao, L., Chen, X., & Huang, Z. (2023). "Adaptive Task Scheduling for Energy Efficiency in Cloud Data Centers". *IEEE Transactions on Cloud Computing*, 11(1), 67-79.

[16] Singh, A., Kaur, P., & Singh, P. (2023). "Energy-Efficient Scheduling Using Improved Genetic Algorithm in Cloud Environment". *Journal of Supercomputing*, 79, 4321-4338.

[17] Zhang, Y., Wang, L., & Liu, X. (2022). "Dynamic Task Scheduling with Energy-Aware Optimization in Cloud Computing". *Journal of Parallel and Distributed Computing*, 170, 1-12.

[18] Kumar, R., Gupta, S., & Das, A. (2023). "Energy-Efficient Task Scheduling Using Firefly Algorithm in Cloud Computing". *Cluster Computing*, 26, 2791-2803.

[19] Li, Y., Chen, Z., & Wang, W. (2023). "Energy Efficient and QoS-Aware Task Scheduling in Cloud Computing Using Deep Reinforcement Learning". *IEEE Transactions on Network and Service Management*, 20(1), 243-258.

[20] Wang, X., Zhao, J., & Hu, L. (2022). "Optimizing Energy Consumption in Cloud Data Centers via Task Scheduling". *Journal of Cloud Computing: Advances, Systems and Applications*, 11, 55-67.

[21] Nair, V., & George, S. (2023). "A Survey on Energy-Efficient Task Scheduling in Cloud Computing: Trends and Directions". *IEEE Access*, 11, 17356-17370.

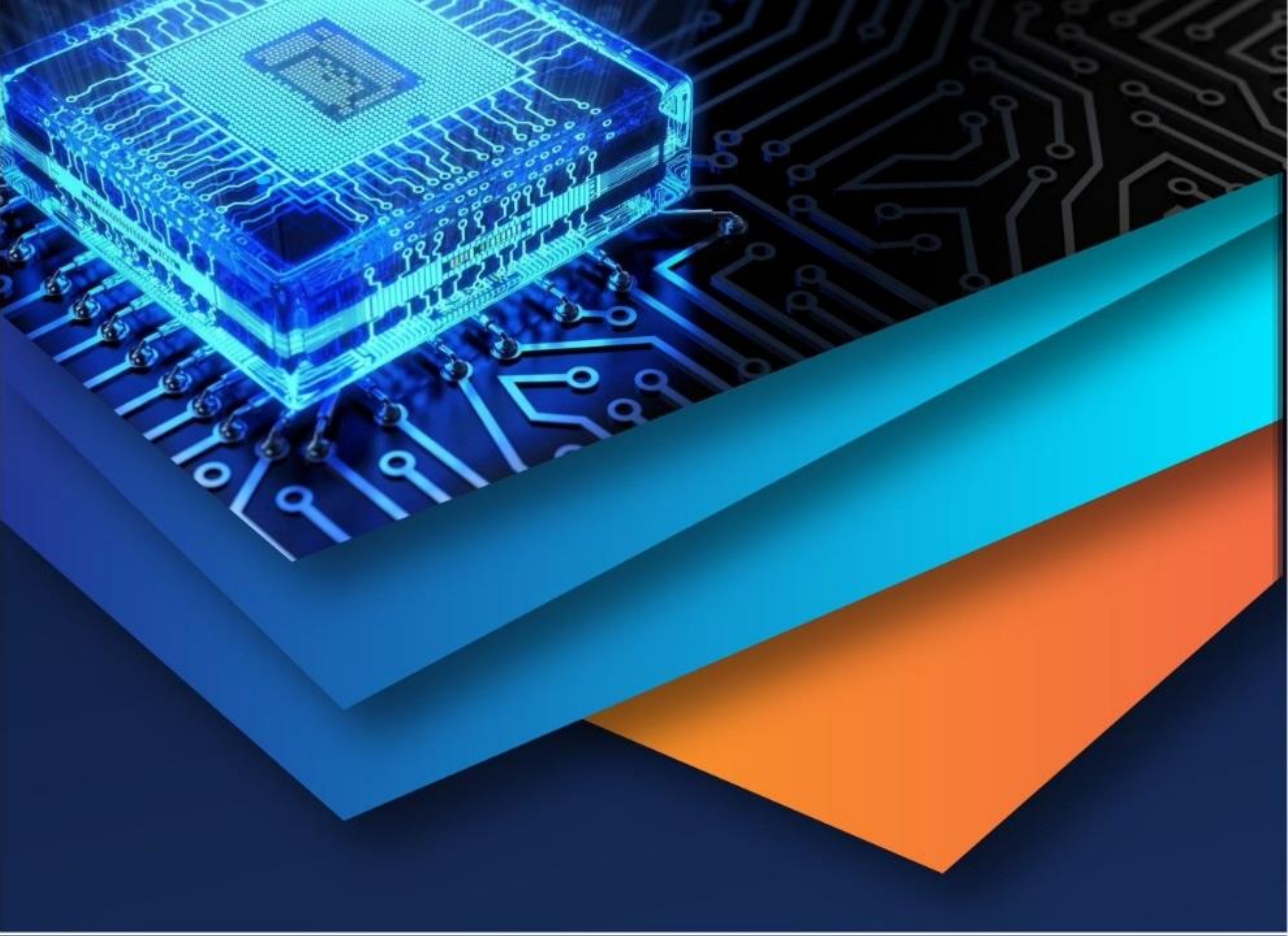
[22] Patel, R., Desai, M., & Shah, P. (2022). "Energy-Efficient Heuristic Algorithms for Task Scheduling in Cloud Computing". *Sustainable Computing: Informatics and Systems*, 35, 100679.

[23] Tan, W., Liang, H., & Huang, R. (2023). "Energy-Aware Task Scheduling for Cloud Computing Using Multi-Objective Optimization". *Computers & Operations Research*, 151, 106080.

[24] Liu, H., Wu, G., & Xu, Y. (2022). "An Energy-Aware and QoS-Guaranteed Task Scheduling Algorithm for Cloud Data Centers". *IEEE Transactions on Sustainable Computing*, 7(3), 678-689.

[25] Kaur, J., Singh, K., & Verma, A. (2023). "Energy-Efficient Task Scheduling Using Ant Colony Optimization in Cloud Environment". *Procedia Computer Science*, 205, 543-550.

[26] Xu, J., Feng, Y., & Wang, S. (2022). "A Comprehensive Survey on Energy-Efficient Task Scheduling in Cloud Data Centers". *Journal of Systems Architecture*, 132, 102788.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY