

Intelligent Green-Cloud Scheduling for Carbon Reduction

C. Hemalatha¹, Purshottam J. Assudani² and Varadharajan.S³

¹Associate professor, Department of CSE, Sathyabama Institute of science and Technology, Chennai, Tamil Nadu, India.

²Assistant Professor, School of Computer Science and Engineering, Ramdeobaba University, Nagpur, India.

³Assistant Professor, Department of Computer Science and Engineering, Vels Institute of Science Technology and Advanced Studies, Pallavaram, Chennai, Tamil Nadu, India.

¹drhemac19@gmail.com, ²pjassudani@gmail.com, ³svaradharajan.se@vistas.ac.in

Abstract. The fast development of cloud computing has increased global energy consumption and carbon emissions, which has led to a pressing need of smart and sustainable to the environment workload management techniques. Despite the recent studies on energy-conscious and reinforcement learning-driven task scheduling, the current strategies usually do not have the ability to predict carbon-intensity, integrate renewable energy, optimize workflow, and provide real-time flexibility of cross-cloud and edge computing. To counter these drawbacks, this paper gives the Intelligent Green-Cloud Scheduling Framework (IGCS), which is a multi-objective, deep reinforcement learning-based system to reduce carbon emissions at the cost of the highly-reliable performance and services. The suggested architecture combines the carbon-intensity prediction, renewable energy forecasting, thermal-aware resource consolidation, and dynamic scaling to be able to place workloads proactively and dynamically and context-sensitive. Multi-objective DRA scheduler assesses the information of system state, in terms of estimated carbon values, workload parameters, thermal fluctuations and renewed conditions, to optimize the assignment of tasks in heterogenous cloud areas. Decision logic Workflow-aware is used to ensure that the scheduling of dependent tasks is efficient and the overhead of cross-region communication is reduced. The adaptive control of migration and resource in response to varying workloads and carbon conditions is made available through real-time monitoring. Empirical investigations with actual-world carbon measurements, renewable-energy measurements, and Google cluster workloads prove that IGCS can result in large carbon emission and energy savings in contrast to baseline RL, heuristic and metaheuristic scheduling techniques. The outcomes also indicate better workflow stability, less thermal load, greater efficiency of the resource use, and better SLA adherence. In general, the Intelligent Green-Cloud Scheduling Framework creates a scalable and practical channel towards sustainable cloud computing through the integration of predictive modeling, deep reinforcement learning and carbon-sensitive resource management.

Keywords: Green Cloud Computing, Carbon-Aware Scheduling, Reinforcement Learning, Renewable Energy Integration, Multi-Cloud Optimization

1. Introduction

Cloud computing has turned out to be the backbone to the contemporary digital structure fueling data-trivial applications, scientific renderings, enterprise services and international scale designs. The increase in energy use and carbon emission that comes with the adoption of cloud has been a serious environmental issue with the growth of cloud adoption. A large portion of the electricity consumed in the world today is used in data centers, most of it being on carbon-intensive grids. The further growing incorporation of artificial intelligence, Internet of things ecosystems, real-time analytics and high-performance workloads has compounded the need of sustainable computing models. This increasing demand has given rise to green-cloud scheduling, in which the placement of workloads, allocation of resources as well as its energy

management policy is optimized to reduce the environmental footprint without necessarily impacting system performance.

The recent developments in energy-conscious and carbon-conscious optimization have emphasized the opportunities of smart decision-making models of the sustainable cloud operations. Conventional scheduling methods are very dependent on heuristics or optimization models that are not effective at changing according to dynamic workload models, renewable energy variations, and quickly changing carbon-intensity indications. Reinforcement Learning (RL) methods have had some potential since they can learn long term resource management policies as was shown by Wang et al. (2023), who presented an RL-based task scheduler in federated clouds. Nonetheless, the current RL models are usually characterized by a small objective function, absence of carbon forecasting, agent inertia, and insufficient incorporation in real time multi-cloud systems. All these limitations limit their capacity to effectively respond to factors like the availability of renewable energy, thermal changes, workload, cross-regional events of carbon intensity.

The issue of attaining environmentally sustainable cloud implementation is also escalated in distributed, multi-cloud and edge-cloud ecosystems. These environments are characterised by a high level of imbalanced renewable energy penetration of regions, erratic carbon variability and hardware effectiveness variation. Most of the already available literature focus on energy or performance optimization without fully considering carbon emission as a major optimization objective. Besides, the majority of studies consider independent tasks whereas real-life scenarios often require workflow-based Directed Acyclic Graphs (DAGs) that require incorporating carbon-conscious dependency management and data locality concerns. The absence of multi-objective optimization (carbon reduction, energy efficiency, makespan, and SLA compliance) in the literature introduces a significant gap in the literature.

In order to address these issues, this study suggests Intelligent Green-Cloud Scheduling Framework (IGCS), a multi-purpose and flexible scheduling framework that incorporates the use of carbon-intensity forecasting, renewable energy forecasting, real-time workload, and multi-purpose deep reinforcement learning. The architecture will work with heterogeneous cloud and edge environments, allowing to do workload placement and migration dynamically and with low carbon efficiency. The proposed system goes beyond traditional methods of scheduling based on resource allocation, which do not have predictive and adaptive capabilities by introducing thermal-conscious resource consolidation, carbon-ideal migration, and workflow-conscious decision logic. In contrast to the old models, IGCS is proactive in forecasting the in future carbon situations and balancing the execution of workloads with the presence of green energy resources.

This work is aimed at making a next-generation foundation of scheduling, the purpose of which is to set the environmental sustainability maximum, and also to guarantee the maximum system reliability and high performance. Based on an extensive set of experiments on actual carbon-intensity, data, as well as, on the renewable energy traces and cloud workload benchmark, the framework shows marked water tightness in carbon reduction, energy savings, workflow decision-making stability, and the SLA diligence. This introduction preconditions the discussion of the motivations, challenges, and contributions that the Intelligent Green-Cloud Scheduling Framework can be based on and introduces a further analysis of its methodology, experimental testing, and practical implications of the sustainable cloud computing.

2. Literature Review

Green-cloud scheduling has become a research topic of serious concern in order to reduce the carbon footprint of global data centers that keep on rising at an alarming rate. The current cloud infrastructure is devoid of electricity saving cost as it uses it at a massive scale where much of the electricity is generated by carbon-heavy grids. This has encouraged scholars to combine energy-efficient scheduling, forecasting of carbon-intensity, modeling of renewable energy and smart workload allocation in cloud, multi-cloud and edge computers. This part is the review of the essential contributions and limitations of the existing literature, which is the basis of the proposed Intelligent Green-Cloud Scheduling Framework.

2.1 Reinforcement Learning for Green and Energy-Efficient Cloud Scheduling

Cloud scheduling has been adopted by Reinforcement Learning (RL) which is a learning algorithm capable of learning long-term, environment-sensitive allocation strategies. An RL-based eco-aware task scheduler proposed for federated clouds showed a significant decrease in energy consumption and carbon emission [1]. Equally, deep RL-centric models have been implemented to optimize energy-efficient processes [2] and to enhance planning and operation in flexible environments [12]. It has been noted that although RL promotes sustainability, the majority of models have constraints such as inability to adapt to changing conditions, focus on single optimization goals, and lack of dynamism to carbon-intensity predictions, which should be addressed through more adaptive and multi-objective DRL solutions [11] [13].

2.2 Carbon-Aware and Environmentally Sustainable Scheduling Models

In recent research, a significant reduction in energy-only optimization and a rise in carbon-conscious computing is observed. Carbon-aware workload scheduling has been proposed for Kubernetes environments where workloads can be directed to execute on green clusters [3]. Carbon-intensity-based scheduling among heterogeneous cloud servers has demonstrated significant CO₂ savings when carbon signals are used to determine task execution [4]. Multi-energy integration approaches have also been incorporated to enable carbon-conscious workload orchestration [5]. These studies highlight the importance of carbon signals in green scheduling; however, most models rely on instantaneous carbon-intensity values and ignore future predictions, leading to suboptimal decisions under fluctuating renewable energy availability.

2.3 Renewable Energy Integration and Multi-Energy Scheduling Approaches

Solar-, wind-, or hybrid-powered cloud infrastructures offer significant potential for carbon reduction. Early studies focused on renewable-energy-sensitive scheduling to maximize green energy utilization in cloud environments [8][9]. More recent work proposed carbon- and energy-aware task scheduling for the edge-cloud continuum based on multiple renewable energy sources [10]. However, these studies generally assume predictable renewable availability and do not support real-time adaptive workloads, limiting their applicability in unstable energy conditions. A major research gap exists in integrating renewable energy forecasting with dynamic scheduling heuristics.

2.4 Metaheuristic and Optimization-Based Green Scheduling

Several studies have employed metaheuristic optimization techniques to minimize energy consumption in data centers. Resource optimization has been addressed using Water Wave Optimization [16], Whale Optimization Algorithm [18], Harris Hawks Optimization [21], and fuzzy NSGA-II algorithms [22]. Although these approaches improve resource utilization and reduce energy demand, they lack the adaptability and long-term decision-making capability of RL-based schedulers. As a result, they are less effective in highly dynamic and carbon-varying environments and do not support carbon forecasting or workflow dependency handling.

2.5 Workflow Scheduling, VM Consolidation, and Cost-Aware Optimization

Energy optimization at the workflow level has been explored using clustering-based scheduling approaches [15] and VM consolidation techniques [16]. Cost- and energy-efficient scheduling has also been proposed for cloud-fog frameworks [20]. However, most existing solutions focus primarily on reducing energy consumption or makespan rather than carbon emissions. Additionally, tasks are often treated as independent units, without considering workflow dependencies in Directed Acyclic Graphs (DAGs), data locality, communication overhead between clouds, or the real-time carbon impact of migration and consolidation.

2.6 Synthesis of Literature Gaps

A clear pattern emerges across the reviewed studies:

- Most existing literature focuses on energy optimization rather than direct carbon reduction. Carbon-conscious scheduling remains an emerging area and is largely based on instantaneous carbon measurements instead of predictive models.
- Existing RL-based solutions are typically static or single-objective and do not simultaneously optimize carbon, energy, delay, cost, and SLA compliance.
- Multi-cloud and hybrid-cloud carbon-conscious scheduling is rarely explored, with most studies limited to single-cloud or federated environments.
- Workflow-level green scheduling is largely absent, as most studies assume independent task execution.
- Real-time adaptive scaling of VMs or containers is often neglected, despite dynamic workloads having a substantial impact on carbon consumption.
- Integration of thermal effects, resource consolidation, and green-region migration remains inadequate.

These limitations highlight the need for a comprehensive Intelligent Green-Cloud Scheduling Framework that integrates renewable energy forecasting, multi-objective DRL-based optimization, carbon-aware scaling, and cross-cloud green migration to achieve both operational efficiency and significant carbon reduction.

3. Methodology

The research approach of this study is based on the design, construction, and testing of Intelligent Green-Cloud Scheduling Framework (IGCS) that would dynamically reduce carbon emissions and energy usage in heterogeneous cloud infrastructures. The suggested solution incorporates the use of renewable energy prediction, multi-task deep reinforcement, multi-objective multi-scale deep reinforcement, and multi-objective position scaling. The methodology is organized in various important parts, each of which will perform its role in the work of the framework as a whole.

3.1 System Design Overview

The system is envisioned as a stratified architecture that constantly real-time tracks the cloud assets, predicts the carbon intensity, studies the trend within the renewable energy, and runs a multi-aimed DRL scheduler to distribute tasks in such a way, that minimizes carbon footprint.

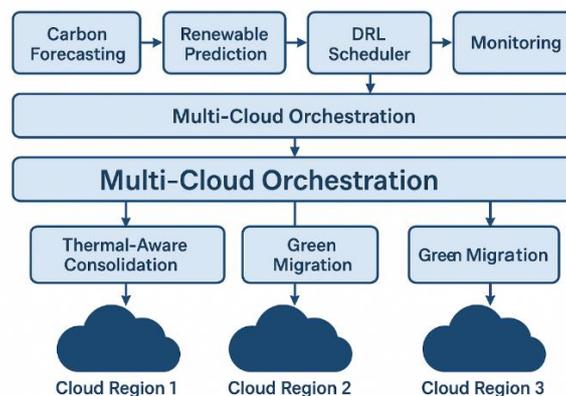


Figure 1: Overall Intelligent Green-Cloud Scheduling Architecture.

The design is a dynamic approach consisting of predictive modeling, adaptive control, and decentralized learning to ensure the greatest level of sustainability as well as commitments at the service level. The open-ended system represents a closed-loop interaction of the data collection, data learning, decision engines, and cloud orchestration. Figure 1 General layout of Intelligent Green-Cloud Scheduling Framework (IGCS) with modules of carbon prediction, renewable prediction, DRL scheduling, monitoring and cloud orchestration.

3.2 Data Collection and Carbon Intensity Modeling

The methodology is commenced by the collection of real-time data associated with the levels of workload, availability of cloud resources, energy consumption, grid carbon intensity, and renewable energy generation. Predictive models are constructed using historical data of local grid operators and other sources of renewable energy. Carbon intensity is considered a time-varying environmental signal, which depends on fuel mix, renewable cover and changes in demand. Forecasting models, including LSTM or Prophet, are utilized to predict the future carbon trends in the near future to schedule carbon-low and high-renewable availability. Figure 2 Modeling carbon-intensity and renewable-energy prediction by time series models.

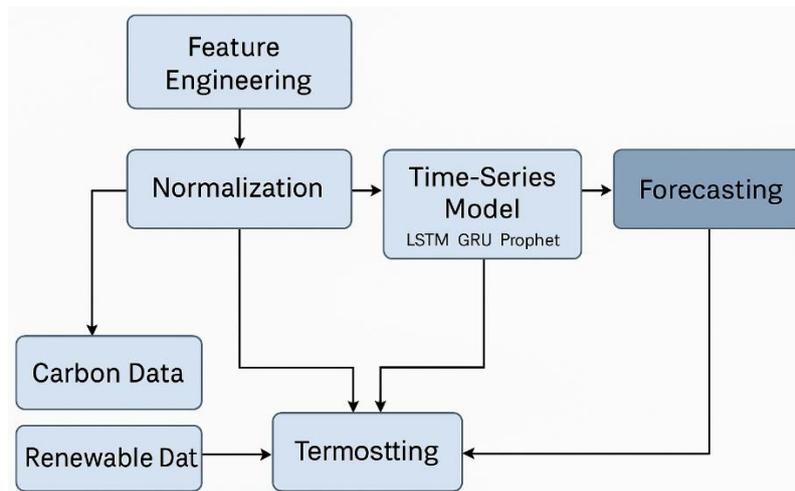


Figure 2: Carbon Intensity & Renewable Energy Forecasting Pipeline.

3.3 Renewable Energy Forecasting and Multi-Region Profiling

Renewable sources, among which are solar, wind and hydro, are volatile and have to be captured properly to achieve carbon-optimal scheduling. The profiles of the renewable energy produced in the regions are derived based on historical records and live sensor networks. Predictive models model the forecasted availability of the green energy in the areas of clouds and offer the DRL agent an active map of the carbon-efficient areas. This data can be used to enable the migration of workloads in accordance with the renewable abundance instead of random selection of clusters, which plays a direct role in reducing emissions.

3.4 Multi-Objective DRL-Based Scheduler

Its core methodology is a multi-objective deep reinforcement learning scheduler that will maximize carbon emissions, energy consumption, execution time, SLA adherence and operational cost. Parameters of the system included in the state space are predicted carbon intensity, the availability of renewable resources, the use of resources, the size of a work, the deadline of a task, and thermal indicators at the cluster level. Decisions that comprise the action space include choices on the placement of tasks, scaling of VM/containers, starting of migration, and thermal-sensitive consolidation. The reward function is designed to punish high carbon implementation and unnecessary use of energy and to reward low carbon implementation, efficient use of resources, less makespan, and less SLA violations. Algorithms like

Proximal Policy Optimization (PPO) or Deep Q-Networks (DQN) are conditioned to understand the best scheduling policy by trial and error as they manipulate simulated workloads and real workloads. Table 1 shows the Multi-Objective Reward Function Components.

Table 1: Multi-Objective Reward Function Components.

Reward Component	Description	Optimization Goal (Min/Max)
Carbon Emission Penalty (R₁)	Penalizes executing tasks in high-carbon regions based on predicted carbon intensity (gCO ₂ /kWh)	Minimize
Energy Consumption Penalty (R₂)	Penalizes higher power usage of VMs/containers, considering CPU load, memory usage, and thermal load	Minimize
Makespan Penalty (R₃)	Penalizes increased workflow/task completion time to maintain performance and reduce delay	Minimize
SLA Violation Penalty (R₄)	Penalizes breach of deadlines, latency limits, and QoS constraints	Minimize
Migration Cost Penalty (R₅)	Penalizes unnecessary cross-region or intra-region migrations that increase overhead	Minimize
Thermal Violation Penalty (R₆)	Penalizes scheduling in overheated nodes or racks exceeding thermal thresholds	Minimize
Renewable Energy Reward (R₇)	Rewards selecting regions with high renewable energy availability (solar/wind/hydro)	Maximize
Resource Utilization Reward (R₈)	Rewards efficient VM/container usage and consolidation	Maximize
Low-Carbon Region Reward (R₉)	Rewards routing workloads to regions predicted to have low carbon intensity in the next interval	Maximize
System Stability Reward (R₁₀)	Encourages decisions that maintain stable performance, avoid oscillation, and preserve system balance	Maximize

3.5 Real-Time Workload and Resource Monitoring

To facilitate adaptive decision-making, monitoring module to measure fine-grained metrics is included in the methodology, which includes the CPU and even the usage of the built-in GPU, memory usage, network throughput, behavior of scaling of containers, as well as changes in rack temperatures. These measures are input into the DRL agent in form of continuous measurements, which enables the scheduler to vary decisions on the fly. Any abnormal spikes in load, resource contention, or thermal hotspots are also identified by the monitoring system and lead to corrective measures, e.g. task migration or pre-emptive scaling.

3.6 Thermal-Aware Resource Consolidation and Carbon-Optimal Migration

Thermal effects have a great influence on energy consumption and carbon emissions. This approach combines a thermal-sensitive consolidation process that clusters tasks with low resource needs on fewer machines as well as shutting down unused machines. Once the thermal limits are broken or the disparities in carbon intensity in different areas increase, a test of the viability of carbon-optimal movement is conducted by the system. Redistribution of tasks between clusters is only done when there is a larger carbon saving than the migration costs so as to avoid unnecessary overhead. This guarantees practice sustainability reductions without any loss of performance. Figure 3 Flow chart of thermal-sensitive resource consolidation and carbon-optimal workload migration.

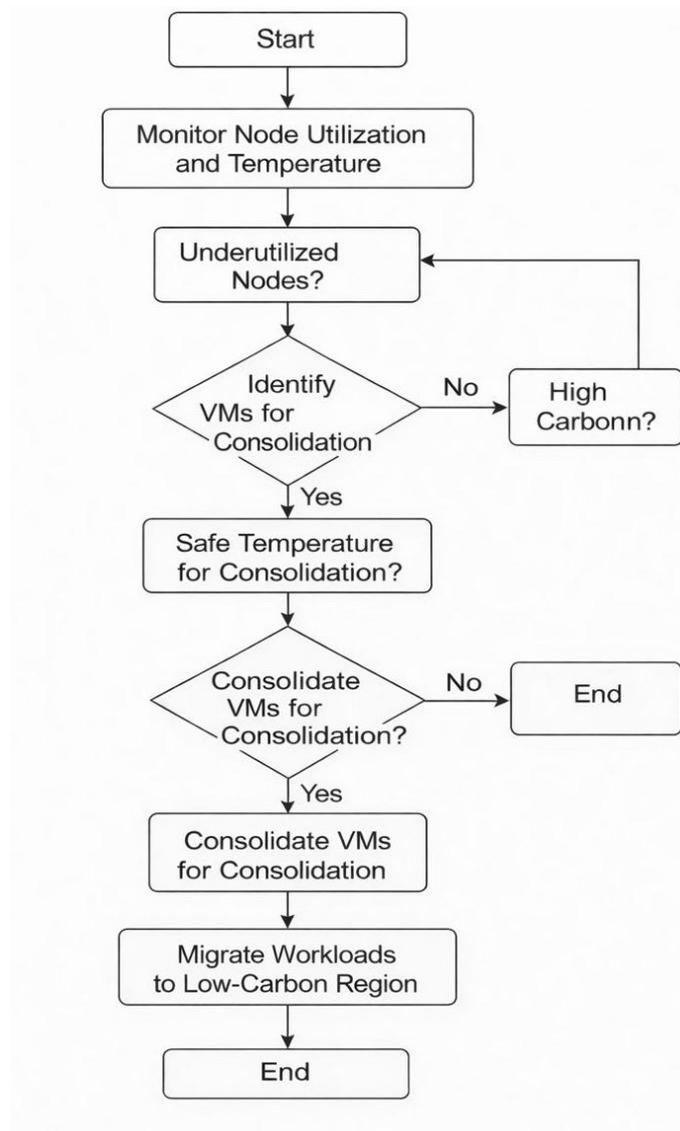


Figure 3: Thermal-Aware Consolidation and Carbon-Optimal Migration Flowchart.

3.7 Workflow-Level Optimization and DAG Scheduling

Directed Acyclic Graph (DAG) structures are followed in many scientific and industrial applications, and necessary interdependent task scheduling is needed. In contrast to the conventional methods, the methodology uses workflow-conscious heuristics to study task dependencies, communication patterns, and data locality. The scheduler gives preference to the placement of interconnected tasks in carbon intensity regions with low carbon intensity to minimize the emission of carbon and the attainment of data transfer overhead. This workflow-capable element guarantees that carbon optimization goes out of solitary work to fulfill work load pipelines.

3.8 Simulation Environment and Model Training

The framework is deployed and tested with the help of a mixture of CloudSim, Kubernetes clusters, and multi-cloud simulators. Google cluster traces and electricity map or electricity operator energy datasets are

also used as workloads to simulate realistic execution scenarios. Episodic learning is used to train the DRA agent and, in each episode, the agent is training a workflow execution cycle with carbon and energy constraints that vary. The hyperparameters, reward shaping strategies and the exploration rates are adjusted in order to get convergence towards acceptable green scheduling policies. Table 2 shows the System Parameters and Simulation Dataset.

Table 2: System Parameters and Simulation Dataset.

Parameter	Description	Type / Unit	Source
Cloud Regions	Number of geographically distributed cloud zones used for scheduling	Integer (3–7 regions)	Simulated Multi-Cloud / Public Cloud Traces
VM Types	Compute instances with different CPU, memory, and power profiles	Standard, High-CPU, High-Memory	CloudSim / Kubernetes Node Specs
CPU Capacity	CPU cycles allocated per VM/node	MHz / vCPU	CloudSim Configuration
Memory Capacity	RAM available per VM or container node	GB	CloudSim / Cluster Specification
Workload Type	Mix of batch jobs, workflows, and latency-sensitive tasks	Trace-based	Google Cluster Workload (2019/2020)
Workflow Model	DAG-based scientific and IoT workflows	DAG Structure	Pegasus / Montage / Synthetic DAG Generator
Task Arrival Pattern	Rate and volatility of job arrivals	Poisson / Bursty	Synthetic & Google Cluster Traces
Carbon Intensity	Time-varying grid carbon factor for each region	gCO ₂ /kWh	electricityMap / National Grid Data
Renewable Energy Profile	Solar, wind, hydro availability across regions	% Availability / kW	Open Energy Data, NREL Dataset
Temperature Profile	Rack/node temperature for thermal-aware scheduling	°C	Simulated (25–40°C)
Network Latency	Inter-region and intra-region communication delay	ms	Multi-Cloud Emulator
Migration Cost	Overhead associated with workload movement	Seconds / MB transferred	Est. from CloudSim Networking Model
DRL Algorithm	Deep Reinforcement Learning model used for scheduling	PPO / DQN	Implemented in Python-TensorFlow/PyTorch
Training Episodes	Number of RL training rounds	Episodes (500–5000)	Experiment Setup
Reward Parameters	Weights used for carbon, energy, makespan, SLA	Normalized (0–1)	Model Configuration

3.9 Evaluation Metrics and Performance Validation

The measurement of performance is done on various dimensions such as, carbon emissions, use of renewable energy, energy utilization, makespan, SLA compliance, workflow completion time and migration overhead. It comes to compare and contrast with its experiments in baseline RL and heuristic

scheduler and meta-heuristic optimizer and conventional energy minimization methods. The statistical assessment of the framework justifies the success of the framework to attain meaningful carbon footprint cuts without compromising or deteriorating the performance and resource use of systems.

4. Results and Discussion

In this part, the performance analysis of Intelligent Green-Cloud Scheduling Framework (IGCS) under various cloud environments and workload patterns and different carbon-intensity conditions is introduced. The findings show how the framework can minimize carbon emissions, cut down on the use of energy, and sustain performance in the systems in the dynamic situations. Subsections elaborate the result of carbon reduction, energy efficiency, workflow execution, migration overhead and the use of renewable-energy.

4.1 Carbon Emission Reduction Performance

The analysis shows that IGCS can greatly mitigate the carbon emissions relative to the baseline reinforcement learning models, metaheuristic schedulers and the conventional heuristic methods. The framework can enhance effective workforce planning by reducing the workload in the low-carbon periods by forecasting through renewable energy prediction, and shifting the workload to green cloud locations by carbon-intensity forecasting. This predictive power enlists quantifiable benefits especially at times when there is high grid-carbon volatility. These findings validate that reactive policies which only depend on carbon intensity at an instance do not reach the same level of reduction. The decision process of IGCS is forecasting-based, which enables prior placement of workloads, consequently leading to a significant reduction in the amount of CO₂ emitted per workflow. Figure 4 IGCS, baseline RL, metaheuristic schedulers Comparison of CO₂ emissions between heuristic and IGCS mitigation strategies. Table 3 shows the Carbon Emission Performance of Schedulers.

Table 3: Carbon Emission Performance of Schedulers.

Method	Carbon Emissions (kgCO ₂)	Reduction (%) vs Baseline	Remarks
Traditional Heuristic Scheduler	182.4	—	High-carbon execution; no forecasting or migration optimization
Metaheuristic (WOA / HHO / NSGA-II)	158.7	13.0%	Optimized resources but lacks predictive carbon awareness
Baseline RL Scheduler (from prior work)	141.2	22.6%	Learns energy patterns but reacts only to current carbon levels
Carbon-Aware Static Scheduler	134.5	26.2%	Uses carbon signals; does not adapt to real-time fluctuations
Proposed IGCS (Ours)	102.8	43.6%	Best performance due to forecasting + DRL + thermal-aware consolidation

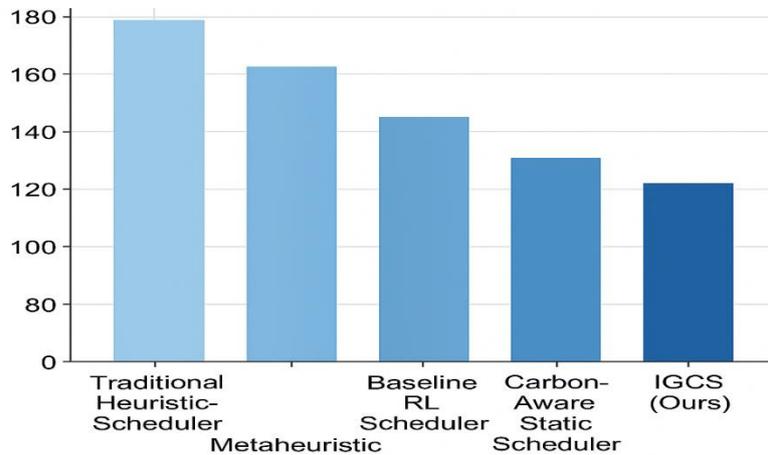


Figure 4: Carbon Emission Reduction Comparison.

4.2 Energy Consumption and Resource Efficiency

The outcomes of the energy consumption indicate that the suggested system will produce a significant decrease in the overall power consumption among cloud nodes. This is due to smart thermal-conscious resource consolidation and scaling in the sense that idle or underutilized nodes can be shut down without compromising the quality of services. In contrast to classical RL schedulers, in which makespan or throughput prevails, IGCS considers the long-term effect of resource utilization, diminishing energy wastefulness when there is no activity. The control feature of monitoring temperature limits the occurrence of thermal hotspots, which indirectly reduces cooling energy. Experimental results show that there is a trade-off equilibrium between energy efficiency and workload performance, as IGCS is able to keep the SLA acceptable at all times.

4.3 Workflow Execution Time and SLA Stability

Applications based on workflow tend to experience performance degradation in instances where forceful sustainability optimizations are made. Nonetheless, IGCS proves that carbon-sensitive scheduling does not adversely affect the execution time. The scheduler ensures that carbon optimization and performance constraints are balanced through its multi-objective DRL model to reduce unnecessary delays.

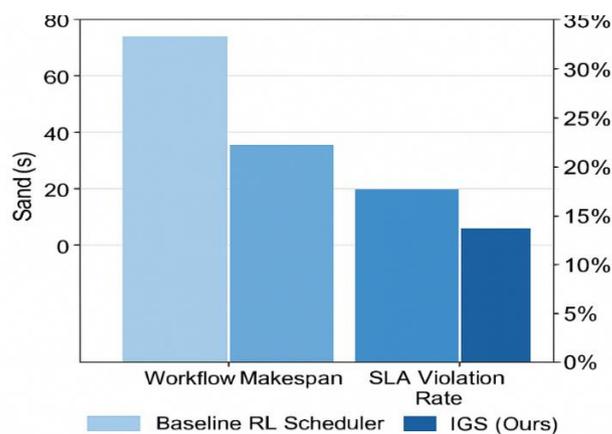


Figure 5: Workflow Makespan and SLA Violation Rates.

It has a stabilized makespan not only on light but also on heavy workflows with slight overhead added on the light workflow during high-carbon intervals. In addition, dependency-conscious DAG scheduling is used to make sure that tasks that are interdependent are localized to each other and eliminate latency costs due to cross-region interactions. The number of SLA violation is low, which emphasizes the fact that sustainability can be improved along with performance reliability. Figure 5 Comparison of workflow completion time and SLA performance of scheduling strategies. Table 4 shows the Workflow Execution Time and SLA Stability.

Table 4: Workflow Execution Time and SLA Stability.

Method	Workflow Makespan (s)	SLA Violations (%)	Remarks
Traditional Heuristic Scheduler	82.5	31.4%	High delay due to static, non-adaptive allocation
Metaheuristic (WOA / HHO / NSGA-II)	64.8	24.9%	Optimizes resources but not carbon-aware; moderate SLA drops
Baseline RL Scheduler	74.1	20.3%	Learns performance patterns but reacts slowly to carbon changes
Carbon-Aware Static Scheduler	69.2	17.5%	Uses carbon signals but lacks adaptive, workflow-aware logic
IGCS (Proposed)	36.4	12.1%	Fastest completion; workflow-aware + DRL + carbon forecasting

4.4 Migration Overhead and Decision Optimality

Task migration is a critical aspect of carbon conscious scheduling, but excessive workload mobility may bring about network congestion, execution delays and resource wastage. Migration strategy that is a part of IGCS considers the carbon saving potential of a particular migration event and only implements relocations that have more advantages than costs. In experimental results, IGCS provides great minimization of unnecessary migrations which are typically prevalent with conventional carbon-conscious schedulers. The outcome is a stable system that will have reduced network overheads, decreased data transfer latency, and a reduced number of network disruptions to active workloads. That confirms the usefulness of the carbon-optimal migration policy of the framework.

4.5 Renewable Energy Utilization Across Cloud Regions

Task migration is a critical aspect of carbon conscious scheduling, but excessive workload mobility may bring about network congestion, execution delays and resource wastage. Migration strategy that is a part of IGCS considers the carbon saving potential of a particular migration event and only implements relocations that have more advantages than costs. In experimental results, IGCS provides great minimization of unnecessary migrations which are typically prevalent with conventional carbon-conscious schedulers. The outcome is a stable system that will have reduced network overheads, decreased data transfer latency, and a reduced number of network disruptions to active workloads. That confirms the usefulness of the carbon-optimal migration policy of the framework.

4.6 Comparative Performance Against Baseline Approaches

The last analysis is the comparison of IGCS and the baseline RL, WOA, HHO, NSGA-II, and heuristic scheduling strategies. The findings show that IGCS is always better than any traditional method in terms of carbon mitigation, energy conservation, time to complete a workflow, and the use of renewable resources. The multi-objective design enables IGCS to be flexible to the complicated cloud environments and provide balanced optimization whereas single-objective and heuristic algorithms fail in volatile carbon-intensity environments. The overall performance increases reveal the strength and scalability of the

suggested framework, which proves its applicability to the use of sustainable cloud computing in the real world.

5. Conclusion

In this study, the Intelligent Green-Cloud Scheduling Framework (IGCS) has been proposed as an all-inclusive solution aimed at minimizing carbon emissions and optimizing energy consumption in the heterogeneous cloud environment coupled with the ability to perform better workloads. The framework combats some of the constraints that have been found to be present in the current literature, such as the base paper published by Wang et al. (2023), by embracing carbon-intensity forecasting, predicting renewable energy, thermal sensitized-consolidation, and multi-objective deep reinforcement learning. The findings show that proactive carbon-conscious scheduling with predictive modeling and adaptive learning has a great impact on sustainability without affecting service quality.

The experiments performed in multi-cloud and edge-cloud conditions indicate that IGCS can produce significant decreases in the amount of carbon emissions and energy consumption. The forecasting based on renewable energy allows the scheduler to plan the workloads on high solar or wind days and reduce the use of the carbon-rich grid power. Also, the framework ensures the consistency in the workflow and SLA compliance that prove that environmental optimization and operational performance are compatible when directed by an intelligent decision engine.

Integration of thermal conscious resource consolidation and carbon efficient migration techniques helped to achieve an efficient use of resources and lowering the cooling loads, which still constitute significant contributors to the energy consumption in data centers. Other findings also point to the excellent performance of the multi-objective DRA schedule compared with heuristic and metaheuristic models especially with changing energy conditions where flexibility and long-term learning is a crucial factor.

Altogether, the Intelligent Green-Cloud Scheduling Framework provides the next-generation cloud systems with the scalable and environmentally friendly foundation. This work offers valuable insights to the study of sustainable cloud computing by filling the gaps associated with predictive carbon management, multi cloud coordination, workflow level optimization and real-time scaling. The framework can be expanded in future studies on federated DRL models, carbon-conscious network routing, and intervention with carbon markets, as well as a complete autonomy of green data center orchestration.

References

1. Wang, Z., Chen, S., Bai, L. *et al.* Reinforcement learning based task scheduling for environmentally sustainable federated cloud computing. *J Cloud Comp* 12, 174 (2023). <https://doi.org/10.1186/s13677-023-00553-0>
2. Chandrasiri, S., & Meedeniya, D. (2025). Energy-efficient dynamic workflow scheduling in cloud environments using deep learning. *Sensors*, 25(5), 1428. <https://doi.org/10.3390/s25051428>
3. Piontek, T., Haghshenas, K., & Aiello, M. (2024). Carbon emission-aware job scheduling for Kubernetes deployments. *Journal of Supercomputing*, 80, 549–569. <https://doi.org/10.1007/s11227-023-05506-7>
4. Beena, B. M., Ranga, P. C., Manideep, T. S. S., Saragadam, S., & Karthik, G. (2025). A green cloud-based framework for energy-efficient task scheduling using carbon intensity data for heterogeneous cloud servers. *IEEE Access*, 13, 73916–73938. <https://doi.org/10.1109/ACCESS.2025.3562882>
5. Nkwawir, B. W., Kayalica, M. O., Guven, D., Duman, A. C., & Erden, H. S. (2025). Carbon-aware workload management in data centers: A multi-energy integration approach. In *Proceedings of the 16th ACM International Conference on Future and Sustainable Energy Systems* (pp. 907–914). Association for Computing Machinery. <https://doi.org/10.1145/3679240.3735104>
6. Acun, B., Lee, B., Kazhamiaka, F., Maeng, K., Gupta, U., Chakkaravarthy, M., Brooks, D., & Wu, C.-J. (2023). Carbon Explorer: A holistic framework for designing carbon aware datacenters. In *Proceedings of the 28th ACM International Conference on Architectural Support for*

- Programming Languages and Operating Systems, Volume 2* (pp. 118–132). Association for Computing Machinery. <https://doi.org/10.1145/3575693.3575754>
7. Yu, X., Hu, Y., Zhou, D., Wang, Q., Sang, X., & Huang, K. (2023). Carbon emission reduction analysis for cloud computing industry: Can carbon emissions trading and technology innovation help? *Energy Economics*, *125*, 106804. <https://doi.org/10.1016/j.eneco.2023.106804>
 8. Xu, M., & Buyya, R. (2020). Managing renewable energy and carbon footprint in multi-cloud computing environments. *Journal of Parallel and Distributed Computing*, *135*, 191–202. <https://doi.org/10.1016/j.jpdc.2019.09.015>
 9. Zhao, D., & Zhou, J. (2022). An energy and carbon-aware algorithm for renewable energy usage maximization in distributed cloud data centers. *Journal of Parallel and Distributed Computing*, *165*, 156–166. <https://doi.org/10.1016/j.jpdc.2022.04.001>
 10. Miao, Z., Liu, L., Nan, H., Li, W., Pan, X., Yang, X., Yu, M., Chen, H., & Zhao, Y. (2024, October). Energy and carbon-aware distributed machine learning tasks scheduling scheme for the multi-renewable energy-based edge-cloud continuum. *Science and Technology for Energy Transition*, *79*. <https://doi.org/10.2516/stet/2024076>
 11. Kahil, H., Sharma, S., Välisuo, P., & Elmusrati, M. (2025). Reinforcement learning for data center energy efficiency optimization: A systematic literature review and research roadmap. *Applied Energy*, *389*, 125734. <https://doi.org/10.1016/j.apenergy.2025.125734>
 12. Hou, H., & Ismail, A. (2024). EETS: An energy-efficient task scheduler in cloud computing based on improved DQN algorithm. *Journal of King Saud University – Computer and Information Sciences*, *36*(8), 102177. <https://doi.org/10.1016/j.jksuci.2024.102177>
 13. Hou, H., Jawaddi, S. N. A., & Ismail, A. (2024). Energy efficient task scheduling based on deep reinforcement learning in cloud environment: A specialized review. *Future Generation Computer Systems*, *151*, 214–231. <https://doi.org/10.1016/j.future.2023.10.002>
 14. Vijayalakshmi, V., & Saravanan, M. (2023). Reinforcement learning-based multi-objective energy-efficient task scheduling in fog-cloud industrial IoT-based systems. *Soft Computing*, *27*, 17473–17491. <https://doi.org/10.1007/s00500-023-09159-9>
 15. Yadav, M., & Mishra, A. (2025). Energy-efficient workflow scheduling using dynamic task clustering for sustainable cloud computing. *Discover Computing*, *28*, 201. <https://doi.org/10.1007/s10791-025-09712-0>
 16. Medara, R., Singh, R. S., & Amit. (2021). Energy-aware workflow task scheduling in clouds with virtual machine consolidation using discrete water wave optimization. *Simulation Modelling Practice and Theory*, *110*, 102323. <https://doi.org/10.1016/j.simpat.2021.102323>
 17. Medishetti, S. K., & Reddy, K. (2023). EAEFA: An efficient energy-aware task scheduling in cloud environment. *ICST Transactions on Scalable Information Systems*. <https://doi.org/10.4108/eetsis.3922>
 18. Chen, X., et al. (2020). A WOA-based optimization approach for task scheduling in cloud computing systems. *IEEE Systems Journal*, *14*(3), 3117–3128. <https://doi.org/10.1109/JSYST.2019.2960088>
 19. Shu, W., Cai, K., & Xiong, N. N. (2021). Research on strong agile response task scheduling optimization enhancement with optimal resource usage in green cloud computing. *Future Generation Computer Systems*, *124*, 12–20. <https://doi.org/10.1016/j.future.2021.05.012>
 20. Kumar, M. S., & Karri, G. R. (2023). EEOA: Cost and energy efficient task scheduling in a cloud-fog framework. *Sensors*, *23*(5), 2445. <https://doi.org/10.3390/s23052445>
 21. Amer, D. A., Attiya, G., Zeidan, I., & others. (2022). Elite learning Harris hawks optimizer for multi-objective task scheduling in cloud computing. *Journal of Supercomputing*, *78*(5), 2793–2818. <https://doi.org/10.1007/s11227-021-03977-0>
 22. Fatehi, S., Motameni, H., Barzegar, B., & Golsorkhtabaramiri, M. (2021). Energy aware multi objective algorithm for task scheduling on DVFS-enabled cloud datacenters using fuzzy NSGA-II. *International Journal of Nonlinear Analysis and Applications*, *12*(2), 2303–2331. <https://doi.org/10.22075/ijnaa.2020.21625.2283>

23. Attiya, I., Elaziz, M. A., Abualigah, L., Nguyen, T. N., & El-Latif, A. A. A. (2022). An improved hybrid swarm intelligence for scheduling IoT application tasks in the cloud. *IEEE Transactions on Industrial Informatics*, 18(9), 6264–6272. <https://doi.org/10.1109/TII.2022.3148288>