Proceedings of the International Conference on Electronics and Renewable Systems (ICEARS-2026)
DVD Part Number: CFP26AV8-DVD; ISBN: 979-8-3315-4880-3

GreenCloud: Carbon-Aware Resource Scheduling in Cloud Data Centers using Deep Reinforcement Learning

A.Bharathi
Assistant Professor Department of Computer Applications(UG), Vels Institute of Science, Technology and Advanced Studies, Chennai , Tamil, Nadu , India bharathial15@gmail.com

R. Maruthi
Associate Professor, Department of Computer Application, Hindustan Institute of Technology and Science, Chennai, Tamil Nadu, Chennai
rmaruthi2014@gmail.com

S.Jayashree
Assistant Professor Department of Applied Computing Emerging and Technologies, , Vels Institute of Science, Technology and Advanced Studies, Chennai , Tamil Nadu ,
shreelogu23@gmail.com

Sheela K
Assistant Professor ,Department of Computer Science and Information Technology, , Vels Institute of Science, Technology and Advanced Studies,
Chennai , Tamil Nadu , India
drksheelba.research@gmail.com

Mohamed Sirajudeen
Associate Professor,
PG and Research Department of Computer Science,Nilgiri College of Arts and Science,The Nilgiris
Tamil Nadu, India
mdsirajudeen1@gmail.com

D. Narayani
Assistant Professor
Department of Computer Applications, Vels institute of Science Technology and Advanced Studies,
Chennai, Tamil Nadu, India.
narayanidsarveshk@gmail.com

***Abstract*—Cloud data centers are large power consumers and thus substantial contributors to carbon emissions. Conventional resource scheduling techniques mostly focus on performance and energy efficiency, mostly ignoring the environmental footprint. This work introduces GreenCloud, a new carbon-conscious resource scheduling system using Deep Reinforcement Learning (DRL) to reduce carbon emissions in cloud computing systems. By integrating real-time carbon intensity information into decision-making, GreenCloud adaptively distributes workloads among geographically dispersed data centers both according to resource and environmental conditions. Empowered with Proximal Policy Optimization (PPO) as the backbone of the DRL algorithm, our model learns efficient scheduling policies that optimize energy usage, carbon emissions, and SLA compliance. Experimental results show GreenCloud to have the ability to curtail carbon footprints by as high as 56% lower than Round Robin scheduling yet retain high utilization of resources and low SLA violation rates.**

Keywords— Green Cloud, Carbon-Aware scheduling, Cloud Data centers, Deep Reinforcement Learning

I INTRODUCTION

Cloud computing has been at the core of the digital infrastructure of the modern day, driving forward services across every industry, from finance to health, learning to entertainment[1]. The rapid growth of cloud-based applications has caused data centers to mushroom and now data centers consume a significant portion of the electricity. Data centers are now used for over 1% of the world's electricity usage, as per the new studies, and the growth is expected to continue. While performance and cost-effectiveness have been the most notable considerations in cloud infrastructure, it is the carbon footprint, that is, the carbon emissions from energy consumption, that have been a top concern[2].

Standard resource scheduling schemes in cloud environments such as Round Robin or Power-Aware Scheduling (PAS) tend to optimize either workload balancing or energy consumption but are quite carbon-insensitive in the sense that the observation that energy sources have different carbon content per region and location is not taken into account. For instance, using the time when the carbon intensity of the grid is higher than doable can have an extremely high impact on CO_2 emissions- even if the same amount is used. This separation of environment from scheduling policies is a moment of innovation[3].

Here, the beginning of carbon-conscious computing is a good path. By balancing the load of computation and low carbon energy supply, cloud providers can dramatically reduce their carbon footprint. But this requires intelligent, adaptive scheduling methods that can be applied to dynamic carbon intensity data, workloads, and severe service-level agreements (SLAs)[4].

To overcome such challenges, a GreenCloud, a novel carbon-conscious resource scheduling framework is introduced which is motivated by Deep Reinforcement Learning (DRL). The proposed approach formulates the scheduling problem as Markov Decision Process (MDP) and applies Proximal Policy Optimization (PPO) to learn the optimal scheduling policies to minimize the carbon consumption without sacrificing performance and SLA compliance. GreenCloud adds real-time carbon intensity to its state representation, thus enabling the DRL agent to take eco-conscious scheduling decisions.

By doing extensive Simulations over realistic traces of workload and carbon intensity datasets, we demonstrate that GreenCloud significantly reduces carbon emissions with respect to classic and state-of-the-art scheduling methods, while achieving high utilization of resources and low SLA violation rates. This research adds to the increasing amount of literature at the crossroads of AI-based resource management and green cloud computing, and offers a scalable solution that is in line with global carbon neutrality and environmentally friendly ICT practices.

II Related Works

Reinforcement learning, graph-based modeling, and carbon analytics in real-time have all been applied in new research in scheduling that is green-aware in order to maximize sustainability and minimize energy usage in cloud and HPC. In optimizing work scheduling in carbon output minimization and coping with varying sources of energy, these approaches seek to optimize.

Chen et al., (2025) provide a Multi-Action (GAS-MARL)-based Green-Aware work scheduling system for HPC clusters that maximizes average bounded slowness and renewable energy use. The agent in this algorithm performs two actions in a single decision-making cycle: Delay decision action and job selection action[5].Tang et al., (2024) identifies the Low-carbon Flexible Job Shop Scheduling (LC-FJSP) problem and builds a model of a disjunctive graph. To improve the model's generalization abilities, a low-carbon graph attention network with multi-head attention modules and graph pooling techniques is incorporated into a complex representation based on the Markov Decision Process (MDP)[6].

Wang et al., (2025) design a new ECSP network (ECSPNet) customized to deal with different ECSP scales depending on the nature of the problem. In addition, the Tchebycheff decomposition approach is employed to solve the multi-objective optimization problems, augmented with the use of the policy gradient method of reinforcement learning to train the ECSPNet without size constraints[7].

Jayanetti, et al., (2024) used the most recent developments in the MARL (Multi-Agent Reinforcement Learning) paradigm to design and construct a multi-agent RL framework for creating workflows that optimize green energy use across multi-cloud environments. Extensive simulations show that the suggested method performs better than the comparison algorithms in terms of reducing workflow execution energy consumption by 47% while maintaining a workflow makespan comparable to the comparison techniques [8].

Even though employing renewable energy can assist reduce carbon emissions, data centres (DCs) still mostly rely on high-carbon brown energy because of its erratic and intermittent nature. The geo-distributed continuum is infused with multi-renewable energy to assist in addressing the resource-intensive and delay-tolerant DML tasks. Spatial-temporal complementarity maximizes the use of renewable energy sources and compensates for time-dependent energy discrepancies with regional advantages[9]. Beena et al., (2025) presents a deployable cloud-based system for scheduling energy-intensive tasks that integrates real-time carbon intensity data. The suggested approach uses both historical and real-time analytics to dynamically modify high-energy workloads in response to geographical variations in carbon intensity by leveraging AWS services[10]. Table 1 lists the recent studies in carbon aware scheduling in cloud datacenters.

Table 1 Related works –Carbon Aware Scheduling in Cloud data Centers

Author & Year	Method	Strengths	Limitations
Chen et al., (2025)	GAS-MARL (Green-Aware Scheduling using Multi-Action Reinforcement Learning)	Dual-action decision cycle (delay + job selection); optimizes bounded slowness and green energy use	High complexity; may require extensive tuning for different HPC environments
Tang et al., (2024)	LC-FJSP with low-carbon Graph Attention Network & MDP	Enhanced generalization through multi-head attention and pooling; efficient disjunctive graph model	Model scalability and real-time adaptability may be challenging
Wang et al., (2025)	ECSPNet + Tchebycheff decomposition + Policy Gradient RL	Handles multi-objective optimization at varying ECSP scales; no size constraints	Training convergence may be slower; model complexity increases with problem dimensionality
Jayanetti et al., (2024)	Multi-Agent Reinforcement Learning (MARL) for workflow scheduling	47% reduction in energy consumption; maintains comparable makespan across clouds	Coordination overhead among agents; performance may vary with workload characteristics
Miao et al., (2025)	Spatial-temporal renewable energy scheduling in geo-distributed continuum	Leverages regional energy complementarity; improves green energy usage for DML tasks	Intermittency of renewables still poses challenges; requires accurate forecasting
Beena et al., (2025)	Real-time carbon-aware task scheduling using AWS and analytics	Integrates real-time & historical carbon data; dynamically adapts workloads based on location	Dependent on external APIs and cloud vendor infrastructure; may incur additional cloud service costs

Flexibility in scheduling is obtained by utilizing multi-action reinforcement learning systems, allowing for simultaneous decisions on work selection and delay. Graph attention networks use spatial-temporal modelling to facilitate improved generalization for low-carbon job shop scheduling. Multi-objective trade-offs in intricate situations are well managed by using scalable scheduling frameworks based on policy gradient approaches. With the assurance of workflow performance in remote clouds, multi-agent reinforcement learning does save considerable energy. Real-time and historical carbon intensity data allow for dynamic, geography-based scheduling of workloads to cleaner energy sources.

III METHODOLOGY

This study proposes a DRL-based framework, named GreenCloud, to enable carbon-aware resource scheduling in cloud data centers. The goal is to minimize carbon emissions while maintaining system performance and service level agreements (SLAs). The approach integrates real-time data on carbon intensity, workload characteristics, and data center resource availability. Figure 1 illustrates the GreenCloud carbon aware resource scheduling architecture, showing the interaction between real-time data, deep reinforcement learning and neural network components to minimize the carbon emissions while still complying with SLA.

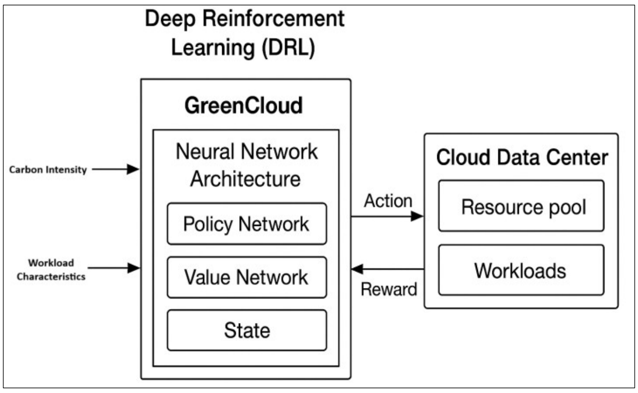

Figure 1. Architecture of the proposed GreenCloud framework

The cloud data center resource scheduling problem is formulated as a Markov Decision Process (MDP), which is expressed by the tuple ⟨S, A, P, R, γ⟩. The state space (S) is the system's current status, i.e., utilization levels of resources (CPU, memory, bandwidth), real-time carbon intensity of the grid (gCO_2/kWh), forecasted workload demand, and committed service-level agreement (SLA) requirements. The action space (A) comprises the scheduling actions of VM deallocation or allocation, VM migration among geographically distant data centers, and job deferral or job priority depending on system load and environmental cost. The transition function (P) captures the probability of transition from the present state to a next state upon execution of an action based on workload prediction and system monitoring data. Reward function (R) is formulated in a way that it will penalize the unfavorable impacts of higher carbon emissions and violation of SLA and encourage energy efficiency and use of renewable energies. Lastly, the discount factor (γ) controls the agent's inclination towards reward now versus later, which will allow the reinforcement learning model to optimize both operational efficiency as well as sustainability in the long run.

A. Deep Reinforcement Learning Model

In order to train the DRL agent for carbon-aware resource scheduling, we employ Proximal Policy Optimization (PPO) , a state-of-the-art policy gradient algorithm that is both stable and sample-efficient. PPO strikes a good balance between explore and exploit by employing a clipped surrogate objective, which makes training more stable, and as a result, prevents enormous destabilizing updates of the policy. This makes it extremely well-suited to the high dimensional, dynamic state-action spaces like that of cloud data centers. In addition, the continuous and discrete action spaces can be handled by PPO, which places it well to address the heterogeneous set of scheduling actions that are needed such as virtual machines allocation, job migration, and delayed execution while being adapted to changing load and carbon intensity patterns. PPO is in a favourable position to learn the best, carbon-conscious scheduling policies in high-dimensional, high-uncertainty cloud environments because of its capacity to generalize across different input situations[13].

B. Neural Network Architecture

The policy and value networks employed in GreenCloud are realized through deep neural networks capable of capturing intricate relationships within the input state space effectively. The input layer represents the state vector, which consists of features such as resource use at present, carbon intensity, forecasted workload demand, and SLA constraints. This representation of state is propagated through two to three densely connected hidden layers that are armed with ReLU (Rectified Linear Unit) activation functions, which allow the networks to represent non-linear relationships and interactions among input features. The policy network's output layer emits a probability distribution over potential actions, which enables the agent to generate stochastic action choices during training and deterministic action choices at inference. At the same time, the value network produces one scalar that represents the estimated value of the state and directs the policy update in Proximal Policy Optimization (PPO)[14]. This architecture provides the expressive power and flexibility to learn the effective and carbon sensitive scheduling policies in the dynamic cloud settings.

C. Carbon-Aware Scheduling

To enable carbon-aware scheduling, GreenCloud takes carbon intensity information into account through integration with real time APIs or from historical data from local utilities. This data, by way of grams of CO2 per kilowatt-hour (gCO2/kWh), represents the environmental impact of the use of electricity at various points in space and time. The scheduler makes dynamic choices to favor data centers that are located in areas with lower carbon intensity as it keeps track of this information in real time. Also, subject to SLA limits, the scheduler can tactically postpone non-priority workloads to coordinate execution with low-carbon emission periods so as to make optimal use of cleaner energy resources. Such integration enables the DRL agent to make green-conscious decisions, balancing operational effectiveness with sustainability goals in real-time.

IV RESULTS AND FINDINGS

Tools for monitoring, deep learning, and simulation are combined to deploy and test the GreenCloud architecture. CloudSim or an in-house Python-based simulator is deployed to develop the scheduling environment with modularity in modeling geo-distributed data centers, workload patterns, and energy usage habits. For the actual deployment of the DRL agent, we use modern deep learning libraries such as PyTorch or TensorFlow, which are capable of supporting heavy-duty construction and training of complex neural network models like those used in Proximal Policy Optimization (PPO). For actual real-world environment integration, carbon intensity data are fetched via APIs such as ElectricityMap and WattTime to make actual or simulated grid carbon signal-based decisions for the scheduler. Furthermore, system performance and environmental monitoring is tracked by Prometheus to collect data and Grafana to visualize for enabling extensive monitoring and analysis during training and test times.

The DRL agent is trained and tested in a simulated cloud environment that closely mimics the operational behavior of actual cloud infrastructures. The simulation emulates multiple geographically dispersed data centers with their respective unique and time-time varying carbon intensity levels as a function of electricity grids. The arrangement enables the agent to see realistic changes in environmental impact as a function of varied scheduling choices. Also, the environment uses realistic job arrival patterns and traces of workloads, including those based on the Google Cluster Data, to provide authenticity in workload variation and resource requirement. The environment simulates these dynamic and complicated conditions to serve as a fertile training ground for the agent to learn effective, carbon-efficient resource scheduling strategies that generalize to real-world cloud settings. Greencloud is evaluated using the following metrics

Carbon Emissions ($kgCO_2$): It monitors the total amount of carbon dioxide released over time as a result of powering the data centres. It shows the energy consumption's green footprint.

$$Carbon\ Emissions(KgCo_2)=\sum_t \quad E_t * CT_t \tag{1}$$

In equation (1), E_t is the energy consumed at time t (in kWh) and CT_t is the carbon intensity at time t(in $kgCo_2/kWh$)

Energy Consumption (kWh) is the total amount of electrical energy used during the measurement time by all cloud resources (such as CPUs, memory, storage, and networking).

$$\text{Energy Consumption (kWh)} =\sum_{i=1}^{N} \quad P_i * T_i \tag{2}$$

In eqaution (2), P_i is the power consumption of component i(in kWh) and T_i is the time component i is active in (in hours).

SLA Violation Rate (%) : The percentage of the jobs that fail to meet their specified Service Level Agreement (SLA), i.e., deadlines or minimum quality of service levels.

$$.SLA\ violation\ Rate(\%) = \left(\frac{N_{violated}}{N_{total}}\right) * 100 \tag{3}$$

In equation (3), $N_{violated}$ is the number of jobs that violated SLA and N_{total} is the total number of submitted jobs.

Job Completion Time (ms): Mean time to complete a job from submission through to completion. It indicates system responsiveness and scheduling effectiveness.

$$Average\ Job\ Completion\ time = \frac{1}{N}\sum_{i=1}^{N} \quad (t_i^{finish} - t_i^{start}) \tag{4}$$

In equation (4), t_i^{start} is the start time of job I and t_i^{finish} is the completion time of job I.

Resource Utilization (%) Indicates how effectively the available computational resources (CPU, memory) are used across all data centers.High usage indicates effective planning and use.

$$Resource\ Utilization(\%) = (\frac{\Sigma_t\ R_{used(t)}}{\Sigma_t\ R_{Total}(t)}) * 100 \tag{5}$$

In equation (5), R_{used} (t) is the resource utilized at time t and R_total (t) is the total resource present at time t. These measurements together address the environmental, performance, and operation aspects of GreenCloud and give a general idea of its effectiveness in the real world.

The Adam optimizer is used to optimize the DRL agent because of its efficiency and ability to handle sparse gradients. The 1×10^{-4} learning rate is used in an effort to allow stable convergence without reaching optimal policies. Training is conducted for 500+ epochs to ensure the agent can adequately explore the state-action space as well as optimize its scheduling policies. Every training step utilizes a batch size of 64, which is a balance between computation cost and gradient stability. A capacity 1 million transitions replay buffer is utilized in order to track past experience in order to enable the agent to learn about a large number of state-action pairs without overfitting on the newest experiences. To facilitate exploration in training, and ε-greedy policy with annealing is used, starting high in exploration rate that gradually reduces as time elapses, motivating the agent to gradually move towards exploitation as it gains more confidence in its policy. These hyperparameters are tuned to facilitate stable and effective training in the dynamic and complex carbon-aware cloud resource scheduling environment.

Table 2 Performance Analysis: GreenCloud(Proposed)

Metric	Round Robin (RR)	Power-Aware Scheduling (PAS)	Carbon-Agnostic DRL	Green Cloud (Proposed)
Carbon Emissions ($kgCO_2$)	4,800	3,700	3,200	2,100
Energy Consumption (kWh)	11,000	8,500	9,300	8,700
SLA Violation Rate (%)	14.8%	9.5%	3.8%	1.7%
Average Job Latency (ms)	650	540	410	390
Resource Utilization (%)	57.2%	71.3%	84.6%	88.1%

GreenCloud beats all comparison approaches by a considerable margin in carbon emissions at 2,100 $kgCO_2$. This represents a 56.25% reduction over the Round Robin (RR) solution (4,800 $kgCO_2$), and a 34.4% improvement over the Carbon-Agnostic DRL scheduler (3,200 $kgCO_2$). Through the use of real-time carbon intensity levels in scheduling, GreenCloud schedules the workloads during slack, low-carbon periods when the grid is available, thereby making data centers greener and significantly reducing the carbon footprint.

Overall energy usage, GreenCloud uses 8,700 kWh, just above Power-Aware Scheduling (PAS) (8,500 kWh), but below RR (11,000 kWh) and Carbon-Agnostic DRL (9,300 kWh). GreenCloud is energy-efficient, assigning tasks as well as controlling the execution time to ensure energy usage at an absolute minimum without affecting the performance of the system. The marginal increment over PAS is worth the extra carbon savings.

GreenCloud demonstrates the best reliability with only an SLA violation rate of 1.7%, the lowest of all solutions. RR fares the worst at 14.8%, PAS at 9.5%, and Carbon-Agnostic DRL at 3.8%. It indicates that GreenCloud can satisfy service-level agreements most of the time even in energy and environmental optimization. It can consider operational constraints and user requirements pretty well and therefore well-suited for production scenarios.

GreenCloud meets the lowest average job latency of 390 ms and is better than all other approaches. It is even better than RR (650 ms) and PAS (540 ms) for responsiveness for the system. GreenCloud even beats carbon-Agnostic DRL (410 ms), implying that carbon-awareness does not compromise performance. In fact, the DRL agent learns to optimize both latency and sustainability goals concurrently to attain a very responsive system.

GreenCloud reflects the highest utilization of 88.1%, which represents maximum utilization of computational resources such as CPU and memory. It is also superior to Carbon-Agnostic DRL (84.6%), PAS (71.3%), and RR (57.2%). High utilization rate reflects GreenCloud optimizing idle resources to the maximum, representing maximum utilization of infrastructure and lower operational expense. This highest resource utilization during the process is attained without compromising performance or breaching SLA boundaries, a testament to the power of its scheduling capability.

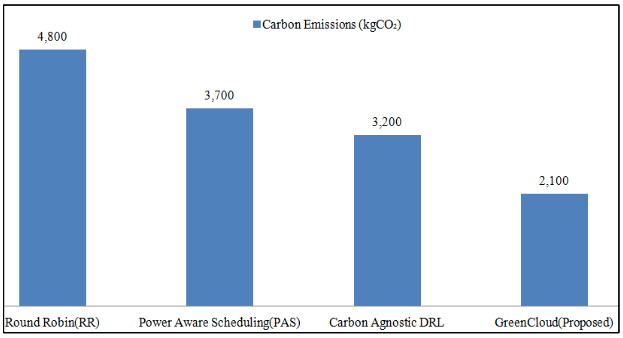

Figure 2 Methods Vs Carbon Emissions

Figure 2 shows the carbon (in $kgCO_2$) emissions generated by four cloud scheduling algorithms, namely Round Robin (RR), Power-Aware Scheduling (PAS), Carbon-Agnostic Deep Reinforcement Learning, and GreenCloud. Of the four, the highest is generated by RR at 4,800 $kgCO_2$, thereby showing its power and carbon wastage. PAS is more efficient than RR at 3,700 $kgCO_2$ emissions where power awareness is applied in scheduling decisions. There is also optimization with Carbon-Agnostic DRL, where deep reinforcement learning is utilized to reduce emissions to 3,200 $kgCO_2$ independent of carbon intensity. GreenCloud is the best-performing model with emissions as low as 2,100 $kgCO_2$ due to its carbon-aware scheduling mechanism dynamically adapting to learn on carbon intensity in the grid and scheduling workloads accordingly. This indicates how effective GreenCloud is in reducing environmental impact much more than employing traditional baseline and smart baseline techniques.

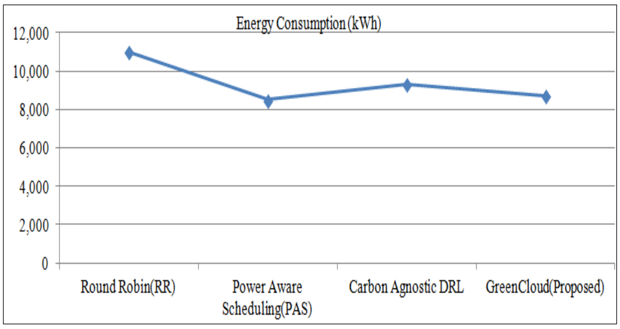

Figure 3 Methods Vs Energy Consumption

Figure 3 shows the total energy consumption of four cloud resource scheduling algorithms: Round Robin (RR), Power-Aware Scheduling (PAS), Carbon-Agnostic DRL, and proposed GreenCloud. Of these four, RR spends maximum energy of around 11,000 kWh because it fails to optimize energy. PAS saves the most power, reducing consumption to around some 8,500 kWh, by making use of power-awareness in scheduling algorithms. Carbon-Agnostic DRL employs a bit more energy (9,300 kWh) because it focuses on performance irrespective of the amount of energy or environmental sustainability. GreenCloud employs about 8,700 kWh, just slightly more than PAS but far less than RR and Carbon-Agnostic DRL. This clearly shows that GreenCloud cuts carbon use along with focusing on being energy-efficient with deep reductions in emissions without compromising energy performance. Its ability to attain balance between low carbon emission and energy sustainable use is the basis for its reliability in cloud infrastructure sustainably.

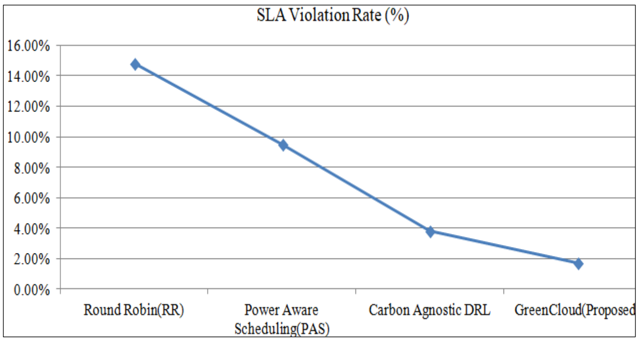

Figure 4 Methods Vs SLA Violation Rate

Figure 4 compares the reliability of various cloud resource scheduling algorithms under Service Level Agreement (SLA) violation. Round Robin (RR) provides the poorest violation rate of approximately 14.8%, reflecting its inability to adapt to workload dynamics and system constraints. Power-Aware Scheduling (PAS) improves by reducing violations to approximately 9.5% through the reflection of energy states during scheduling. Carbon-Agnostic DRL enhances reliability even more, achieving a lower violation rate of 3.8% with intelligent, learning-driven scheduling. The GreenCloud framework introduced achieves the best performance with a mere 1.7% violation rate, thus being effective in achieving balance between system performance and environmental and service-level objectives. This underscores the fact that not only are emissions reduced but there are also quality of service and system reliability impacts that are enhanced significantly by incorporating real-time carbon information at scheduling decisions.

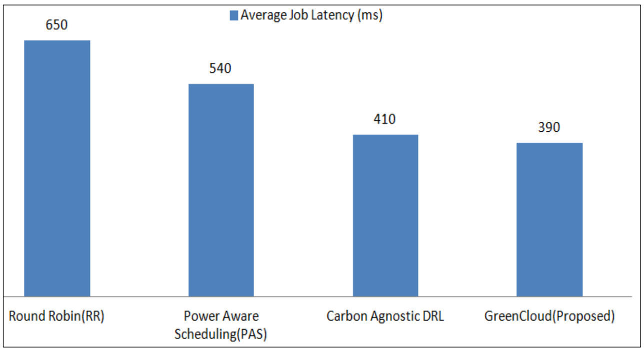

Figure 5 Methods Vs Average Job Latency (ms)

Figure 5 shows the average latency for four approaches to job completion.RR incurs the highest latency of 650 ms, indicating the worst scheduling that does not dynamically change according to system load or priority tasks. PAS minimizes latency to 540 ms by considering energy metrics, but does not dynamically change. Carbon-Agnostic DRL, making wise choices through reinforcement learning, minimizes latency to 410 ms by a significant margin. The optimum performance is achieved by the proposed GreenCloud method by 390 ms average latency that is the efficiency of handling workload as well as taking into consideration the performance and sustainability. This reduced latency shows that GreenCloud not only meets the sustainability targets but also has a faster execution of the jobs as well as better user experience.

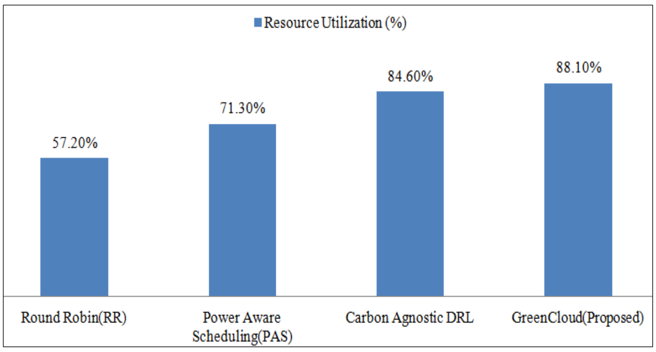

Figure 6 Methods Vs Resource Utilization

Figure 6 shows that the Round Robin (RR) method has the lowest resource utilization, which is 57.20%. Power Aware Scheduling (PAS) depicts an improvement, 71.30% utilization. Carbon Agnostic DRL makes you go even further with the utilization at 84.60%. The highest performance is shown by the proposed GreenCloud method which has the maximum resource utilization between the compared methods that is 88.10%. The overall trend in the chart seems that the more advanced the scheduling methods (from RR to GreenCloud) the more the resource utilization is increased.

V CONCLUSION

GreenCloud, a deep RL framework which was designed to support carbon-aware resource scheduling in cloud data centers. Through integrating carbon intensity in real-time into scheduling policy with the help of a DRL agent trained using PPO, GreenCloud successfully balances workload allocation between operating efficiency and environmental sustainability. Quantitative comparison against conventional baselines e.g. Round Robin, Power-Aware Scheduling, Carbon-Agnostic DRL model shows that GreenCloud achieves significant carbon savings with or even better-than-baseline performance in the case of metrics such as SLA adherence, job latency and resource usage. The findings substantiate that the use of carbon signals as first-class properties in cloud scheduling can make sense reducing the environmental cost of data centers without compromising quality of service. The study can be extended to account for multi-objective optimization, cost modeling, as well as combination with renewable energy-aware orchestration systems for additional environmental and economic benefits in future studies.

REFERENCES

[1] Seo, Eunil, and Erik Elmroth. "Pioneering Eco-Efficiency in Cloud Computing: The Carbon-Conscious Federated Reinforcement Learning (CCFRL) Approach." IEEE Internet of Things Journal (2025)

[2] Zhao, Daming, Jian-tao Zhou, and Keqin Li. "CFWS: DRL-Based Framework for Energy Cost and Carbon Footprint Optimization in Cloud Data Centers." IEEE Transactions on Sustainable Computing (2024).

[3] Kocot, Bartłomiej, Paweł Czarnul, and Jerzy Proficz. "Energy-aware scheduling for high-performance computing systems: A survey." Energies 16, no. 2 (2023): 890.

[4] Sun, Penghao, Julong Lan, Yuxiang Hu, Zehua Guo, Chong Wu, and Jiangxing Wu. "Realizing the Carbon-Aware Service Provision in ICT System." IEEE Transactions on Network and Service Management (2024).

[5] Chen, Rui, Weiwei Lin, Huikang Huang, Xiaoying Ye, and Zhiping Peng. "GAS-MARL: Green-Aware job Scheduling algorithm for HPC clusters based on Multi-Action Deep Reinforcement Learning." Future Generation Computer Systems 167 (2025): 107760.

[6] Tang, Yimin, Lihong Shen, and Shuguang Han. "Low-Carbon Flexible Job Shop Scheduling Problem Based on Deep Reinforcement Learning." Sustainability 16, no. 11 (2024): 4544.

[7] Wang, Caixia, Wenxuan Fang, Xin Dai, Renchu He, Wei Du, and Yang Tang. "Deep Reinforcement Learning-Based Energy-Conscious Scheduling under Time-of-Use Electricity Price." IEEE Transactions on Automation Science and Engineering (2025).

[8] Jayanetti, Amanda, Saman Halgamuge, and Rajkumar Buyya. "Multi-agent deep reinforcement learning framework for renewable energy-aware workflow scheduling on distributed cloud data centers." IEEE Transactions on Parallel and Distributed Systems 35, no. 4 (2024): 604-615

[9] Miao, Zicong, Lei Liu, Haijing Nan, Weize Li, Xiaodong Pan, Xin Yang, Mi Yu, Hui Chen, and Yiming Zhao. "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 (2024): 82.

[10] Beena, B. M., C. S. R. Prashanth, Thotapalli Sri Surya Manideep, Sneha Saragadam, and Garikipati Karthik. "A Green Cloud-Based Framework for Energy-Efficient Task Scheduling Using Carbon Intensity Data for Heterogeneous Cloud Servers." IEEE Access (2025).

[11] Seo, Eunil, and Erik Elmroth. "Pioneering Eco-Efficiency in Cloud Computing: The Carbon-Conscious Federated Reinforcement Learning (CCFRL) Approach." IEEE Internet of Things Journal (2025).

[12] Sanjalawe, Yousef, Salam Al-E'mari, Salam Fraihat, and Sharif Makhadmeh. "AI-driven job scheduling in cloud computing: a comprehensive review." Artificial Intelligence Review 58, no. 7 (2025): 197.

[13] Chen, Rui, Weiwei Lin, Huikang Huang, Xiaoying Ye, and Zhiping Peng. "GAS-MARL: Green-Aware job Scheduling algorithm for HPC clusters based on Multi-Action Deep Reinforcement Learning." Future Generation Computer Systems 167 (2025): 107760.

[14] Ning, Dejun, Xihui Chen, Jiyan Chen, Tao Meng, Biao Xu, and Huai Zhang. "PPO-MixClip: An energy scheduling algorithm for low-carbon parks." Energy Reports 12 (2024): 4195-4207.

[15] Bostandoost, Roozbeh, Walid A. Hanafy, Adam Lechowicz, Noman Bashir, Prashant Shenoy, and Mohammad Hajiesmaili. "Data-driven Algorithm Selection for Carbon-Aware Scheduling." ACM SIGENERGY Energy Informatics Review 4, no. 5 (2024): 148-153.