



CLOUD SUSTAINABILITY AND GREEN COMPUTING

Mahima Pandey¹, Mr. Nilesh Khare², Komal Yadav³

School of Computer Applications, Babu Banarasi Das University, Lucknow, Uttar Pradesh, India

DOI : <https://doi.org/10.5281/zenodo.23587863>

ABSTRACT :

Cloud infrastructure provides elastic and shared computational capacity, yet workload placement can increase electricity demand, operational carbon emissions and service-level risks when resource decisions overlook regional energy characteristics. This paper presents Sustainability-Aware Scheduling and Consolidation (SASC), an interpretable multi-objective framework on cloud sustainability and green computing. SASC first rejects allocations that violate capacity, security or quality-of-service constraints, and then ranks feasible placements by normalized incremental IT energy, carbon cost, expected SLA risk, movement overhead and desirable host utilization. Geographic carbon-intensity differences and facility-efficiency factors are included in the emissions accounting boundary. A power-based energy model, an operational-carbon estimator, utilization safeguards and migration hysteresis support a reproducible simulation design. The proposed evaluation compares SASC with Round Robin, consolidation-oriented Best-Fit and carbon-oblivious energy-aware scheduling using heterogeneous hosts, variable workloads and temporally varying carbon-intensity profiles. Paired workload replays, multiple random seeds, ablation experiments, uncertainty intervals and QoS non-inferiority checks are specified to separate potential environmental gains from performance displacement. Analytical conditions show why the minimum-energy destination need not be the minimum-carbon destination and identify circumstances in which geographical shifting is counterproductive. The contribution is a testable, auditable decision architecture and a complete evaluation protocol rather than an assertion of unmeasured performance improvement. The framework is also translated into procurement and operational indicators for institutions seeking evidence-based sustainable cloud adoption.

Keywords—Cloud sustainability; green computing; carbon-aware scheduling; cloud simulation.

1. INTRODUCTION

Cloud computing pools computation, storage and network services across virtualized infrastructure to deliver resources elastically. This architecture can raise average resource utilization, but its sustainability is not automatic: idle capacity, thermal overhead, workload relocation and latency-sensitive replication may offset efficiency gains. A systematic review by Panwar et al. identifies resource provisioning, consolidation and infrastructure management as interconnected energy-control layers [1]. The anticipatory VM-allocation work of Caglar and Altılar further illustrates why avoiding needless migration is relevant to both electricity demand and execution continuity [2].

Scheduling decisions become more complex when infrastructure is distributed across geographic regions. Multi-objective VM-placement studies combine power, waste and communication delay [3], while federated-cloud research demonstrates that the environmental desirability of a destination depends on its local energy mix as well as its computational characteristics [4]. Thermal-aware resource management adds another constraint: concentrated processing can create cooling demand or hotspots even when it reduces the number of operational servers [5]. Therefore, green cloud scheduling is best framed as a constrained decision problem rather than a stand-alone CPU-utilization maximization exercise.

Sustainability-Aware Scheduling and Consolidation (SASC). The proposed policy couples energy and carbon information with operational safeguards while retaining an explainable weighted-score mechanism. In contrast to approaches that report an energy minimum alone, it evaluates energy, operational carbon, SLA performance, utilization and migration together, with each indicator assigned an explicit measurement boundary.

The primary research question is: Can an interpretable, carbon- and SLA-aware scheduler lower the environmental burden of representative cloud workloads relative to conventional and energy-only allocation without unacceptable service-quality deterioration? Two directional hypotheses guide future testing. H1: SASC reduces IT energy and estimated operational carbon relative to a conventional scheduling baseline across representative load conditions. H2: carbon-aware weighting lowers estimated operational emissions relative to energy-only scheduling while satisfying pre-specified QoS limits. The hypotheses are not presented as confirmed findings.

The paper contributes (i) a documented optimization model with safety constraints; (ii) a decision procedure linking placement, consolidation and geographic carbon information; (iii) a paired, reproducible simulation protocol with meaningful ablations; and (iv) an institutional interpretation of sustainability metrics. Sections 2–3 synthesize prior work and delimit the research gap, Sections 4–6 formalize and evaluate the proposed design, and Sections 7–9 discuss analytical implications, limitations and conclusions.

2. LITERATURE SURVEY

2.1 Workload awareness, VM placement and facility effects

Workload-informed allocation reduces the risk of optimizing for an unrepresentative average request. Sharma et al. combine workflow classification with energy-aware scheduling for geo-distributed cloud systems, emphasizing that workload heterogeneity can materially influence allocation decisions [6]. Patel et al. extend the accounting perimeter toward the cloud–edge continuum, including the impact of devices, communication and supporting infrastructure [7]. At the broker layer, genetic-algorithm-based data-center selection demonstrates that destination choice affects response time and resource efficiency even before carbon is explicitly modeled [8].

The distinction between operational energy efficiency and wider sustainability is developed by Pazienza et al., who emphasize a systems-level carbon-aware view [9]. Network optimization research identifies communication protocols and data movement as additional sources of resource consumption [10]. Hardware-efficiency trajectories provide an important counterpoint: faster or more energy-efficient equipment does not obviate the need for demand-sensitive scheduling, since growing workloads can consume the gains [11]. These findings jointly motivate a cost term for workload movement and a report that distinguishes IT energy from facility-adjusted operational emissions.

2.2 Intelligent and adaptive allocation strategies

Dynamic clustering can co-locate compatible tasks and adapt execution to greener supply conditions [12]. Edge and serverless scheduling approaches make decisions from changing resource states rather than permanent server rankings [13]. Fuzzy or hybrid algorithms combine energy management with resilience [14], while reviews of higher-education cloud adoption show that green-computing outcomes also depend on governance, procurement, capability and post-adoption management [15]. Recent ant-colony/particle-swarm optimization research offers sophisticated energy-oriented scheduling [16], and QoS-aware VM-placement research explicitly treats the energy–service trade-off as multi-objective [17].

These studies advance optimization quality but do not remove the methodological need for a minimal, auditable benchmark. A transparent score that discloses its normalization, weights, exclusion rules and operating assumptions can help determine whether additional algorithmic complexity is justified by verifiable environmental gains. It can also support sensitivity analysis that attributes policy changes to energy, carbon or QoS criteria rather than obscuring those relationships behind a black-box optimizer.

TABLE 1. RESEARCH STREAMS AND IMPLICATIONS FOR THE PROPOSED FRAMEWORK

Research stream	Studies	SASC design implication
Provisioning and VM consolidation	[1]–[3], [5]	Estimate incremental energy; protect overload and thermal thresholds.
Geographic and workload-aware control	[4], [6], [8], [12]	Vary workload profile, destination and carbon intensity.
Whole-system environmental accounting	[7], [9]–[11]	Separate IT electricity, facility overhead and relocation costs.
Adaptive or hybrid optimization	[13], [14], [16], [17]	Use transparent baselines and explicit SLA safeguards.
Institutional sustainable adoption	[15]	Translate metrics into cloud governance and procurement checks.

3. RESEARCH GAP AND DESIGN REQUIREMENTS

The literature reviewed six technically consequential gaps. First, minimizing electrical energy is not equivalent to minimizing associated carbon emissions when grid carbon intensity varies by location or time. Second, placement, networking, cooling and broker decisions are often investigated independently, making it difficult to quantify displacement effects. Third, evaluations do not always report an interoperable set of environmental and QoS indicators. Fourth, aggressive consolidation can harm tail latency or availability. Fifth, a favorable result under one trace or carbon profile need not survive workload, intensity or weight changes. Finally, complex optimizers may be difficult to reproduce or explain in a smaller institutional deployment.

These observations translate into testable requirements: R1, evaluate both energy and operational carbon; R2, reject infeasible destinations before scoring; R3, account for activation, movement and facility overhead; R4, compare policies on identical replayed demand; R5, test workload and carbon heterogeneity; R6, disclose weights, normalization and experimental uncertainty; and R7, generate recommendations that an educational institution can use without assuming direct control of its provider’s hardware.

4. SYSTEM MODEL AND SUSTAINABILITY FORMULATION

4.1 Entities, feasibility and scheduling horizon

Let I be a set of incoming tasks or VM requests and J the set of candidate hosts and data-center regions. Each request i specifies CPU demand c_i , memory demand m_i , bandwidth demand b_i , a deadline or response-time target d_i , and any geographic or security restrictions. A binary decision x_{ij} indicates whether request i is assigned to candidate j . A placement is eligible only if CPU, memory and bandwidth capacity remain feasible, policy constraints are satisfied and predicted service risk does not exceed a predefined limit. Geographic movement is disallowed for data that cannot legally

or technically leave its authorized region.

$$x_{ij} \in \{0,1\}; \sum_j x_{ij} = 1; x_{ij} = 0 \text{ for every infeasible candidate.} \quad (1)$$

Scheduling occurs on arrival and at controlled consolidation epochs. For an ongoing service, the benefit of changing placement must exceed both migration cost and a hysteresis margin; this limits oscillation when carbon signals fluctuate. Execution windows are constrained by task deadlines, and irreversible interactive requests are not postponed merely because lower-carbon electricity may become available later.

4.2 Energy and emissions accounting

For host h , let $P_{\text{idle},h}$ and $P_{\text{peak},h}$ denote idle and peak power (W), and let $u_h(t)$ be modeled CPU utilization in $[0,1]$. A simple, clearly identified first-order host-power model is used for comparability. The numerical integration step Δt is measured in hours. The model is a protocol assumption and may be replaced by a validated non-linear power curve when available [1], [5].

$$P_h(t) = P_{\text{idle},h} + [P_{\text{peak},h} - P_{\text{idle},h}]u_h(t). \quad (2)$$

$$E_{\text{IT}} = (1/1000) \sum_r \sum_t P_h(t) \Delta t \quad [\text{kWh}]. \quad (3)$$

For region r at time t , let $CI_r(t)$ be grid-electricity carbon intensity ($\text{kgCO}_2\text{e/kWh}$), and let $PUE_r(t)$ be a defined facility-overhead multiplier, if available. Estimated location-based operational carbon is calculated from region-attributed IT energy, adding separately metered transfer or migration electricity where the boundary requires it. Facility adjustments must not be applied twice to electricity already measured at the facility meter.

$$C_{\text{op}} = \sum_r \sum_t [E_{\text{IT},r}(t) \times PUE_r(t) \times CI_r(t)] + C_{\text{move}} \quad [\text{kgCO}_2\text{e}]. \quad (4)$$

Where PUE cannot be supported by a data-center value, results should show unadjusted IT energy and a clearly labeled range or sensitivity scenario rather than inventing a facility measurement. PUE is a facility efficiency indicator under ISO/IEC 30134-2:2026 [18]; neither PUE nor the carbon intensity of consumed electricity alone fully captures the cradle-to-grave impact of servers. The Green Software Foundation's Software Carbon Intensity (SCI) approach also recognizes embodied hardware emissions and the need for a defined functional unit [21]. Accordingly, C_{op} in Equation (4) is an operational estimate, not a full life-cycle carbon footprint.

The incremental-energy estimate ΔE_{ij} is the modeled increase in total IT energy associated with placing i on j , including whether a sleeping host must start or an extra host must remain active. A policy that charges only the dynamic CPU component while ignoring the idle-on cost could spuriously prefer highly distributed placement. Similarly, carbon accounting must distinguish the electricity used by a new task from migration traffic or application-level duplication.

4.3 Normalized multi-criteria decision score

Within the feasible candidate set F_i , five quantities are computed: normalized marginal energy $\hat{E}_{ij}$, normalized operational carbon $\hat{C}_{ij}$, normalized predicted SLA risk $\hat{R}_{ij}$, normalized movement cost $\hat{M}_{ij}$, and normalized desirable utilization $\hat{U}_{ij}$. A deterministic min-max transform maps each cost to $[0,1]$ across current feasible candidates; when all candidate values are equal, the corresponding normalized quantity is set to zero. This avoids division by zero and makes the decision rule reproducible. Utilization is rewarded only up to a safe utilization target; any candidate above the hard capacity or overload threshold remains infeasible.

$$S_i(j) = wE \hat{E}_{ij} + wC \hat{C}_{ij} + wS \hat{R}_{ij} + wM \hat{M}_{ij} - wU \hat{U}_{ij}. \quad (5)$$

$$wE, wC, wS, wM, wU \geq 0; \quad wE + wC + wS + wM + wU = 1. \quad (6)$$

For each request, SASC chooses $\arg \min_j S_i(j)$, breaking numerical ties by a declared deterministic criterion (e.g., least additional network distance, followed by host identifier). Negative signs represent benefits and do not imply negative energy consumption. Since energy and carbon terms are correlated, their weights must be interpreted as policy preference parameters rather than independent physical quantities. Reporting both objectives separately prevents improvement in the composite score from being confused with environmental improvement.

To anchor sensitivity experiments, the protocol proposes three candidate weight profiles in the order (energy, carbon, SLA risk, movement, utilization): balanced (0.25, 0.30, 0.25, 0.10, 0.10), energy-priority (0.45, 0.10, 0.25, 0.10, 0.10) and carbon-priority (0.10, 0.45, 0.25, 0.10, 0.10). These are design settings that require preregistration and sensitivity checks, not empirically tuned optima. All alternatives remain subject to the same hard QoS constraints.

4.4 Anti-thrashing and consolidation safeguards

Energy-sensitive consolidation may reduce active hosts but can induce a cycle of repeated migrations. Under SASC, a candidate migration from current placement a to destination b is accepted only when b remains feasible, its predicted emissions and energy include the migration overhead, and the score improvement exceeds δ for at least one defined stabilization interval. A minimum host sleep duration, safe post-consolidation utilization ceiling and maximum migrations per task can further reduce oscillation. These guards should be switched off in ablation experiments to quantify the importance of operational stability.

5. SASC ALGORITHM AND SYSTEM ARCHITECTURE

The proposed architecture includes five modules. The workload monitor records incoming demand, deadlines and observed host load. The state repository maintains capacity, power, region and carbon-intensity information. The feasibility filter enforces infrastructure and governance constraints. The scoring engine estimates the five decision components and chooses an eligible destination. Finally, a measurement layer exports timestamped energy, carbon, latency, utilization and movement metrics. Each module records its input values and decision outcome so another researcher can independently recompute a placement score.

TABLE 2. SASC DECISION PROCEDURE (CONCEPTUAL PSEUDOCODE)

Step	Procedure
1	Read task requirements and current host/region telemetry at decision time t .
2	Construct eligible set F_i using CPU, memory, bandwidth, regional and SLA constraints.
3	If F_i is empty, apply documented queue/rejection policy; record the QoS consequence.
4	For each j in F_i , estimate ΔE_{ij} , carbon cost, SLA risk, movement cost and safe utilization.
5	Normalize components across F_i using fixed, logged rules; compute Equation (5).
6	Select lowest-score placement; apply deterministic tie-breaking and dispatch.
7	At consolidation epochs, evaluate migrations only if overhead and hysteresis safeguards pass.
8	Record decisions and the separately measured environmental/QoS outcomes.

If the number of candidate placements is $|J|$ and each candidate can be evaluated in constant time from cached telemetry, one placement decision requires $O(|J|)$ work plus monitoring and feasibility-check costs. This tractability is useful as a transparent benchmark; it does not demonstrate superior optimality relative to evolutionary or reinforcement-learning methods. Carbon forecasts, workload estimation and migration overhead models can later be substituted without changing the evaluation boundary.

6. REPRODUCIBLE EXPERIMENTAL PROTOCOL

6.1 Simulation setting and workload construction

The CloudSim Plus or an equivalent Java-based cloud simulator. CloudSim Plus is a modular research-oriented simulator that supports extensible host, VM and resource-allocation policies [18]. The experimental environment should include multiple regions, heterogeneous hosts, queueing and execution behavior, region-specific carbon-intensity profiles and power models. Scripts and pinned dependency versions should capture the complete model configuration.

Public Google Borg cluster traces may be used to derive distributions of requests and resource use; the maintained ClusterData2019 collection contains workloads from eight Borg cells during May 2019 [21]. A trace-derived subset must document sampling, preprocessing, excluded fields and mapping of trace units into simulation resources. In addition, seeded workload generators should create low, medium and high demand, different burst levels and task-length distributions. A single workload profile cannot establish robustness.

The same workload arrivals, region state series, random seeds, host capacities, deadline assumptions and measurement periods must be replayed under every competing policy. A warm-up interval and explicit shutdown/idle accounting should be included so that a policy is not rewarded merely by moving unfinished demand beyond the observation window. Task rejection, postponed work and incomplete requests must remain visible in the results.

6.2 Policy comparators and experimental design

The minimum comparison set is: Round Robin (conventional capacity-filtered placement), Best-Fit or utilization-oriented consolidation without carbon signals, energy-only placement minimizing modeled incremental IT energy, and the proposed SASC policy under three predefined weight profiles. All policies use identical hard capacity and governance restrictions. Where their SLA handling differs, that difference must be recorded rather than concealed by the allocation implementation.

A practical factorial design crosses three workload levels (low/medium/high), three carbon contrasts (weak/moderate/strong) and six policy configurations (three baselines plus three SASC weight profiles). With 30 paired stochastic repetitions for each workload-carbon scenario, this forms a planned 1,620 policy runs; the exact run count may be amended once computational cost is known. Paired seeds are the appropriate experimental units for uncertainty estimation because every policy experiences the same replay within a replicate.

TABLE 3. PREDEFINED EXPERIMENTAL FACTORS AND REPORTING

Factor or control	Planned alternatives / specification	Primary purpose
Scheduling policy	Round Robin; Best-Fit; energy-only; SASC balanced; SASC energy-priority; SASC carbon-priority	Policy comparison
Load intensity	Low; medium; high; equal job-demand replay for all policies	Stress and generality

Carbon contrast	Weak; moderate; strong regional variation; fixed time series within paired run	Value of geographic carbon awareness
Replications	30 matched random seeds per scenario, subject to computational feasibility	Sampling variability
Constraints	Same host limits; SLA thresholds; region permissions; observation horizon	Fair comparison
Ablations	Remove carbon term; remove movement penalty; remove utilization reward; disable migration hysteresis	Attribution of effect

6.3 Outcome metrics and carbon-accounting discipline

Primary outcomes are total modeled IT energy in kWh and estimated location-based operational carbon in kgCO_{2e}, both aggregated over the same completed-work boundary. Secondary outcomes are successful throughput, completion makespan, mean and 95th-percentile response time, SLA violation proportion, rejected or missed tasks, average and upper-tail host utilization, active-host time and number of migrations. A sustainability comparison is invalid if one algorithm executes materially less useful work without that difference being disclosed.

TABLE 4. EVALUATION METRICS AND INTERPRETATION

Metric	Unit / denominator	Interpretation
IT energy	kWh per horizon and per completed request	Direct simulator energy boundary
Operational carbon	kgCO _{2e} per horizon and per completed request	Facility-adjusted, region/time-specific estimate
Mean / p95 response	milliseconds or seconds; completed requests	Service responsiveness and tail behavior
SLA breach rate	breaches / eligible requests × 100%	Non-inferiority safeguard
Throughput / completion	successful tasks / unit time; successful tasks / arrivals	Avoids apparent gains from dropped work
Utilization	mean and p95 of used / provisioned capacity	Consolidation and hotspot risk
Active hosts / migrations	host-hours; count plus estimated migration energy	Infrastructure and transfer overhead

Facility overhead, grid carbon factors and embodied hardware impact must never be collapsed into an unlabeled single figure. The main outcome is an operational estimate; a secondary SCI-style functional-unit measure may be calculated only when workload allocation and embodied-emissions assumptions are adequately documented [19]. When historical average carbon factors are substituted for hourly signals, that resolution limitation should accompany every plotted emissions comparison.

6.4 Statistical strategy and robustness tests

For a primary comparison, define the paired replicate difference $D_s = Y_{\text{SASC},s} - Y_{\text{baseline},s}$ under identical seeds and scenario configuration. Report the mean or median of D_s , its 95% uncertainty interval and the percentage change relative to the paired baseline. A paired t-test may be used when differences meet its assumptions; a Wilcoxon signed-rank test or paired bootstrap provides alternatives when the distribution is unsuitable. To assess multiple policies within the same replay set, use a repeated-measures procedure or a mixed-effects model with workload and carbon factors; adjust selected post-hoc contrasts for multiplicity. The independent unit is the scenario replicate, not each simulated task.

For quality-of-service protection, define in advance an allowable response-time or SLA-breach difference and test whether SASC remains within that bound. Merely observing a non-significant difference does not prove non-inferiority. Secondary sensitivity analysis should vary the weight vector, overload threshold, minimum migration benefit δ , carbon-series temporal resolution, facility multipliers and workload burstiness. Report outcomes that worsen as well as those that improve and show the Pareto trade-off between operational emissions and service quality.

Validation includes unit tests for power integration and carbon conversion, a hand-computed placement example, decision-level logging, reproducible random seeds and independent recomputation of aggregate figures from exported event logs. The final study should archive configuration manifests, trace identifiers, code versions and analysis scripts. Claims based on simulation must remain distinct from metered production-data-center performance.

7. ANALYTICAL IMPLICATIONS AND DISCUSSION

7.1 When carbon-aware placement can differ from energy-aware placement

A useful analytical comparison can be derived without running a simulator. Suppose two feasible placements, A and B, deliver the same useful service. Let e_A and e_B denote the resulting IT electricity, f_A and f_B their PUE multipliers and g_A and g_B the relevant electricity carbon intensities.

Placement B can emit less operational carbon than A even if it uses more IT electricity, provided its cleaner supply outweighs the combined additional electricity and facility overhead. The necessary pairwise inequality is:

$$e_B f_B g_B < e_A f_A g_A$$

$$\Leftrightarrow (e_B/e_A) < (f_A g_A)/(f_B g_B). \quad (7)$$

Equation (7) is an analytical boundary, not an experimental result. It shows why a scheduler that optimizes only IT kWh can rank destinations differently from a carbon-minimizing scheduler. Conversely, when carbon intensities are nearly equal, a cross-region transfer with non-negligible network overhead may increase both electricity use and emissions. Differences in PUE or unmodeled network energy can reverse a superficially attractive decision. This highlights the importance of boundary-complete accounting and strong/weak contrast tests [4], [7], [9].

7.2 Preventing performance displacement and carbon leakage

Green scheduling does not produce a defensible benefit when environmental costs are shifted outside the reporting boundary. Examples include scheduling fewer requests, leaving work queued after the measurement horizon, excluding migration energy, omitting idle-on host consumption and transferring work to a remote region whose electricity or cooling overhead is not counted. A comparison must therefore normalize by completed useful work as well as report total environmental outcomes. It should present QoS and rejection figures adjacent to emissions figures rather than relegating them to secondary appendices.

The selected weighted score is deliberately interpretable but not universally optimal. If the weight on carbon is raised, the chosen region may change even when expected IT energy is flat; if the SLA term or its hard constraint is strengthened, a lower-carbon but distant site may be rejected. Each shift is legitimate only if its cause is measurable. This is why score components, feasible-set size and override events should be part of the exported decision log. A score reduction by itself is not evidence of kWh or carbon reduction.

7.3 Distinguishing expected patterns from verified outcomes

The protocol predicts three outcomes to test rather than presume. First, consolidation may reduce host idle energy most at low or moderate load, while high load can limit feasible consolidation. Second, carbon-aware decisions should differ from energy-only decisions most often when regions have large carbon-intensity contrasts, substantial geographic freedom and manageable communication latency. Third, the movement penalty and hysteresis should reduce destabilizing workload shifts, but may also forego some short-lived carbon opportunities. These are falsifiable expectations: contrary results would help identify inappropriate power curves, poor forecasts, inaccurate carbon factors or an excessively rigid QoS constraint.

The study should emphasize trade-offs rather than one headline improvement. For example, a configuration that yields lower estimated emissions but higher p95 latency beyond an institutional requirement does not satisfy the paper's central objective. Similarly, a policy that improves modeled electricity demand only when migration energy is excluded does not establish a robust green-computing advantage. Reporting such negative or mixed findings is integral to credible multi-objective evaluation.

8. INSTITUTIONAL IMPLICATIONS AND IMPLEMENTATION PATHWAY

For a university, SASC can be interpreted less as a request to alter a cloud provider's private scheduler and more as a model for evidence-based workload placement and vendor selection. Institutions can classify computing jobs into latency-critical services, movable batch workloads and location-restricted research applications. The ability to choose a region or execution window should be granted only to eligible jobs; student-facing applications with strict responsiveness or data-residency requirements remain subject to those constraints. Higher-education cloud-adoption research makes clear that organizational readiness and governance influence whether technical sustainability measures can be adopted effectively [15].

A procurement-oriented dashboard can request comparable evidence on data-center location, reported PUE methodology, electricity carbon factors, metered energy allocation and service-level guarantees. Where the provider publishes only aggregate sustainability claims, an institution should disclose the resulting measurement gaps rather than extrapolate site-specific emissions. A staged implementation could begin with offline comparisons on anonymized workload histories, proceed to a shadow scheduler that makes recommendations without dispatching tasks, and only later evaluate a limited class of delay-tolerant workloads under operational oversight.

The organizational success criterion is not simply that a cloud supplier appears greener on paper. It is that the chosen service delivers the required volume and quality of computation with accountable resource and environmental costs and documented restrictions on movement. Such an approach can connect technical green-computing metrics to institutional reporting without presupposing access to provider-internal telemetry.

9. LIMITATIONS, RESEARCH INTEGRITY AND FUTURE WORK

The framework remains subject to simulation-to-production transfer risk. Linear host-power models approximate, but do not capture, all processor states, memory behavior, storage I/O, network transport, thermal dynamics or renewable-energy curtailment. Historical electricity-intensity data may not represent consequential marginal emissions; the use of averages should be explicitly described as location-based operational accounting rather than a proof of avoided grid emissions. The cooling multiplier may differ within a region and across time. These uncertainties should be probed in a sensitivity envelope.

The SASC score depends on normalization against the current feasible alternatives. Consequently, adding or removing a candidate can alter normalized values and rankings, and workload predictions may introduce forecast error. Geographic transfers may violate confidentiality, sovereignty or latency requirements. Hard constraints are necessary but can reduce environmental flexibility. The framework also omits embodied manufacture and end-of-life

impacts from its main C_{op} measure; any broader life-cycle claim requires additional evidence and hardware allocation assumptions [19]. Future research can add calibrated non-linear power telemetry, real-time carbon signals, cooling co-optimization and explicit network energy models. In a second phase, Pareto-front optimizers or reinforcement-learning schedulers could be benchmarked against the same transparent SASC protocol. A final extension is controlled implementation on a small testbed with metered hosts and observed network latency, followed by independent validation of whether simulation rankings persist in operational cloud conditions.

10. CONCLUSION

Cloud sustainability requires decisions that jointly consider useful computation, infrastructure electricity, electricity carbon intensity and the continuity of cloud services. Sustainability-Aware Scheduling and Consolidation framework that ranks feasible allocations using energy, operational carbon, utilization, SLA risk and movement overhead. The design combines explicit safety constraints, reproducible normalization, migration safeguards and a factorial simulation protocol against conventional, consolidation-oriented and energy-only alternatives. Its analytical contribution clarifies the conditions under which an energy-minimizing location can be carbon-inferior, while its evaluation design specifies how any environmental advantage must be verified without degrading QoS or concealing unfinished workload. The proposed approach is suitable for future simulation and institutional assessment; validation of H1 and H2 awaits execution and transparent reporting of the planned experiments.

REFERENCES

- [1] S. S. Panwar, M. M. S. Rauthan, and V. Barthwal, "A systematic review on effective energy utilization management strategies in cloud data centers," *Journal of Cloud Computing*, vol. 11, art. no. 95, 2022, doi: 10.1186/s13677-022-00368-5.
- [2] I. Caglar and D. T. Altılar, "Look-ahead energy efficient VM allocation approach for data centers," *Journal of Cloud Computing*, vol. 11, art. no. 11, 2022, doi: 10.1186/s13677-022-00281-x.
- [3] A. Gopu et al., "Energy-efficient virtual machine placement in distributed cloud using NSGA-III algorithm," *Journal of Cloud Computing*, vol. 12, art. no. 124, 2023, doi: 10.1186/s13677-023-00501-y.
- [4] Z. Wang et al., "Reinforcement learning based task scheduling for environmentally sustainable federated cloud computing," *Journal of Cloud Computing*, vol. 12, art. no. 174, 2023, doi: 10.1186/s13677-023-00553-0.
- [5] L. Mao, R. Chen, H. Cheng, W. Lin, B. Liu, and J. Z. Wang, "A resource scheduling method for cloud data centers based on thermal management," *Journal of Cloud Computing*, vol. 12, art. no. 84, 2023, doi: 10.1186/s13677-023-00462-2.
- [6] A. P. Sharma et al., "Sustainable energy efficient workflow classification and scheduling in geo distributed cloud datacenter," *Discover Sustainability*, vol. 5, art. no. 128, 2024, doi: 10.1007/s43621-024-00308-0.
- [7] Y. S. Patel et al., "Modeling the Green Cloud Continuum: Integrating energy considerations into Cloud-Edge models," *Cluster Computing*, vol. 27, pp. 4095–4125, 2024, doi: 10.1007/s10586-024-04383-w.
- [8] S. Chowdhury, A. Katangur, and A. Sheta, "Optimization of datacenter selection through a genetic algorithm-driven service broker policy," *Journal of Cloud Computing*, vol. 13, art. no. 156, 2024, doi: 10.1186/s13677-024-00719-4.
- [9] A. Pazienza et al., "A holistic approach to environmentally sustainable computing," *Innovations in Systems and Software Engineering*, vol. 20, pp. 347–371, 2024, doi: 10.1007/s11334-023-00548-9.
- [10] J. Fesl, M. Polak, and D. Sedlak, "TCP-based network communication optimized for green cloud computing," *Cluster Computing*, vol. 28, art. no. 457, 2025, doi: 10.1007/s10586-025-05251-x.
- [11] A. Prieto et al., "Evolution of computing energy efficiency: Koomey's law revisited," *Cluster Computing*, vol. 28, art. no. 42, 2025, doi: 10.1007/s10586-024-04767-y.
- [12] M. Yadav and A. Mishra, "Energy-efficient workflow scheduling using dynamic task clustering for sustainable cloud computing," *Discover Computing*, vol. 28, art. no. 201, 2025, doi: 10.1007/s10791-025-09712-0.
- [13] M. Ghorbani, M. Ghobaei-Arani, and L. Esmaeili, "An energy-conscious scheduling framework for serverless edge computing in IoT," *Journal of Cloud Computing*, vol. 14, art. no. 52, 2025, doi: 10.1186/s13677-025-00780-7.
- [14] M. K. Roberts et al., "A nature-inspired dual phased fuzzy hybrid algorithm for adaptive self-healing and dynamic energy-aware resource management in cloud computing," *Journal of Cloud Computing*, vol. 14, art. no. 62, 2025, doi: 10.1186/s13677-025-00788-z.
- [15] P. Hatta, Hadiyanto, and C. W. Budiyo, "A systematic literature review of sustainable cloud computing adoption in higher education institutions covering architectures, critical factors, and green IT practices," *Discover Computing*, vol. 29, art. no. 266, 2026, doi: 10.1007/s10791-026-10160-7.
- [16] M. S. Sanaj et al., "Energy-Efficient Task Scheduling in Cloud Computing with EcoTaskOpt: A Hybrid Ant Colony and Particle Swarm Optimization Approach," *International Journal of Computational Intelligence Systems*, vol. 19, art. no. 64, 2026, doi: 10.1007/s44196-025-01095-w.
- [17] E. I. Elsedimy, M. Herajy, and S. M. M. Abohashish, "Energy and QoS-aware virtual machine placement approach for IaaS cloud datacenter," *Neural Computing and Applications*, vol. 37, pp. 2211–2237, 2025, doi: 10.1007/s00521-024-10872-1.
- [18] ISO/IEC 30134-2:2026, Information technology—Data centres key performance indicators—Part 2: Power usage effectiveness (PUE), 2nd ed., Jan. 2026. [Online]. Available: <https://www.iso.org/standard/30134-2>
- [19] Green Software Foundation, "Software Carbon Intensity (SCI), ISO/IEC 21031:2024." [Online]. Available: <https://greensoftware.foundation/standards/sci/> (accessed Oct. 14, 2026).

-
- [20] M. C. Silva Filho, R. L. Oliveira, C. C. Monteiro, P. R. M. Inácio, and M. M. Freire, "CloudSim Plus: A cloud computing simulation framework pursuing software engineering principles for improved modularity, extensibility and correctness," in Proc. IFIP/IEEE Int. Symp. Integrated Network and Service Management, 2017, doi: 10.23919/INM.2017.7987304.
- [21] Google, "Borg cluster traces from Google (ClusterData2019)," Google Cluster Data Repository. [Online]. Available: <https://github.com/google/cluster-data> (accessed Oct. 14, 2026).