Journal of Informatics Education and Research
ISSN: 1526-4726
Vol 6 Issue 1 (2026)

Carbon-Aware Resource Management in Cloud, Edge, and AI Platforms: A Comprehensive Survey

Bhavya N[1], Dr. R.K. Bharathi[2]

[1]Assistant Professor in Department of Computer Applications, JSS Science and Technology University,Mysuru District 570006, Karnataka, India.

Mail: bhavya.nk@jssstuniv.in

[2]Professor in Department of Computer Applications ,JSS Science and Technology University, Mysuru District 570006, Karnataka, India

Mail: rkbharathi@jssstuniv.in

Abstract:

The rapid expansion of cloud computing, big data analytics, Internet-of-Things (IoT) platforms, and artificial intelligence has significantly increased the environmental impact of modern computing systems. While numerous studies propose techniques to improve energy efficiency, existing research remains fragmented, focusing on individual components such as data centers, scheduling algorithms, or machine learning models. This paper presents a comprehensive survey of sustainable computing approaches through a layered analysis of carbon-reduction strategies across the computing ecosystem. The reviewed literature is categorized into six operational layers: carbon measurement and accounting, infrastructure optimization, carbon-aware workload scheduling, sustainable data processing, green artificial intelligence, and continuous application services. The survey analyzes the effectiveness, trade-offs, and validation methods of existing techniques and highlights key limitations, including unreliable emission measurement, simulation-based evaluation, and conflicts between performance and sustainability. Results show that isolated optimizations often shift emissions rather than reduce total environmental impact. The study further identifies major research challenges such as lack of standardized carbon metrics, neglect of hardware lifecycle emissions, and absence of cross-layer coordination. Finally, future directions are outlined, emphasizing carbon-aware orchestration, lifecycle-aware system design, and renewable-integrated computing. The findings demonstrate that meaningful carbon reduction requires coordinated system-level optimization rather than isolated improvements within individual computing components.

Keywords- Green computing, carbon-aware scheduling, sustainable cloud computing, green artificial intelligence, energy-efficient data centers, carbon footprint reduction.

1. **Introduction:**

The digitization of the global economy has rendered computing infrastructure an essential substrate of modern civilization yet this dependency carries a mounting environmental cost. Data centres, telecommunications networks, and edge computing nodes now account for a material and growing share of worldwide electricity consumption, with demand accelerating in lockstep with the proliferation of cloud-native services, large-scale AI systems, and always-on digital platforms [1,12,14]. Projections consistently indicate that without structural intervention, the operational carbon intensity of computing will increase substantially over the coming decade, making the sector a significant contributor to global greenhouse gas emissions [11,18]. Green computing the systematic pursuit of reduced environmental impact across computational systems without compromising performance or reliability—has consequently moved from a peripheral concern to a central imperative in both industry and academic research. Initial work in sustainable computing focused narrowly on hardware-level efficiency: optimizing power delivery, improving cooling infrastructure, and consolidating server workloads to reduce idle energy consumption [3]. These interventions, while valuable, addressed only a fraction of a much larger and more structurally complex problem. Today, the environmental footprint of computing extends across a deeply interconnected ecosystem spanning cloud platforms, AI model training pipelines, big data analytics stacks, edge and IoT deployments, and continuously operating services such as cybersecurity monitoring and smart city applications [6,15,16]. A foundational obstacle complicating progress in this domain is the absence of reliable, granular carbon measurement. The majority of cloud providers currently report carbon estimates derived from generalized resource-utilization proxies rather than direct energy telemetry [1,2]. This reliance on coarse-grained accounting models undermines the validity of optimization strategies: without ground-truth emissions data, it is impossible to rigorously evaluate whether a given scheduling or infrastructure decision achieves genuine carbon reduction or merely redistributes demand. Real-time, measurement-grounded carbon accountability is therefore a prerequisite not a complement to effective green computing practice [12,14].

Concurrent research threads have proposed interventions at multiple system layers: carbon-aware workload scheduling and geographic load shifting in cloud environments [5,12]; renewable-powered microgrids and lifecycle-aware hardware provisioning at the infrastructure level [3]; and techniques such as model pruning, federated learning, and transfer learning to curb the outsized emissions generated by large-scale AI training [4,7,9,11,20]. Emerging application domains—IoT platforms, connected urban infrastructure, persistent monitoring systems introduce additional energy obligations that existing frameworks have yet to fully account for [8,10,15]. However, a critical limitation persists across this body of work: its essential fragmentation. Existing studies overwhelmingly optimize a single layer of the computing stack in isolation, without accounting for the cascading effects of those decisions on adjacent layers. Workload migration that reduces server utilization may simultaneously increase network transmission energy; AI efficiency gains achieved through algorithmic compression may be offset by the embodied carbon of the specialized accelerators on which compressed models are deployed [11,14]. Computing systems are vertically integrated stacks—decisions propagate upward and downward and point-level optimization cannot substitute for system-level design.

This survey addresses that structural gap by introducing a layered taxonomy of carbon-reduction techniques spanning the full sustainable computing ecosystem. We treat green computing not as a monolithic engineering challenge but as a multi-layer optimization problem encompassing carbon measurement and transparency, physical infrastructure, workload management, data processing pipelines, AI systems, and application-tier services. Through a synthesis of recent literature, this work aims to (1) categorize and position existing approaches within a coherent layered framework, (2) surface the performance–sustainability trade-offs that cross-layer analysis reveals, and (3) identify the unresolved research gaps that currently prevent the large-scale deployment of genuinely carbon-aware computing systems.

2. Carbon Footprint Measurement and Carbon Awareness

Before carbon emissions can be reduced, they must first be quantified.
A recurring theme across sustainable computing literature is that optimization techniques depend heavily on the accuracy of carbon measurement. However, current computing infrastructures lack standardized mechanisms for measuring the environmental impact of computational workloads. Instead, most platforms rely on indirect estimation models based on resource usage and electricity emission factors rather than direct energy telemetry [1], [2].

2.1 Carbon Accounting in Computing Systems

Carbon footprint in computing systems is generally defined as the amount of greenhouse gas emissions, expressed as carbon dioxide equivalent (CO_2e), produced during the execution of digital workloads. The environmental impact originates primarily from electricity consumption, which depends on both the amount of power used and the carbon intensity of the electrical grid [12].

A commonly adopted framework is the Greenhouse Gas (GHG) Protocol, which categorizes emissions into three scopes:

- **Scope 1:** Direct emissions from owned infrastructure
- **Scope 2:** Indirect emissions from purchased electricity (dominant in cloud computing)
- **Scope 3:** Embodied emissions from hardware manufacturing and disposal

Most cloud computing research focuses only on Scope 2 emissions because they are easier to estimate, while Scope 3 emissions remain poorly understood due to the lack of supply-chain transparency [11], [12].

Carbon emissions are typically computed as:

$$CO2e = Energy\ Consumption\ (kWh) \times Carbon\ Intensity\ (gCO2/kWh)$$

Although simple in theory, obtaining reliable values for both terms is difficult in practice.

2.2 Carbon Reporting by Cloud Service Providers

Major cloud providers supply sustainability dashboards and carbon reports, but these reports are not based on direct measurements of user workloads. Instead, providers approximate emissions by allocating a fraction of total data center energy consumption to users based on resource utilization metrics such as CPU hours, storage, and memory usage [1].

The major limitations identified across the literature include:

- aggregated monthly reports
- absence of real-time carbon data
- proprietary accounting formulas
- lack of access to actual server power usage

Because the internal power usage effectiveness (PUE), cooling overhead, and infrastructure energy distribution are not exposed to users, independent verification of emissions is impossible [1], [2].

Table 1 — Comparison of Cloud Carbon Reporting Approaches

Aspect	**Native CSP Reports**	**Third-Party Tools**	**Research Measurement Tools**
Data Source	Provider usage logs	Estimated models	Hardware telemetry
Time Granularity	Monthly or daily	Daily/weekly	Real-time
Accuracy	Low (estimated allocation)	Medium (modeled)	High (direct measurement)
Access to Hardware Power	No	No	Yes (limited environments)
Transparency	Proprietary	Partial	Open
Optimization Support	Limited	Moderate	High

(based on findings in [1], [2], [12])

2.3 Carbon Intensity and Grid Awareness

Electricity does not have a constant environmental impact.
The same workload can produce different emissions depending on when and where it is executed because power grids vary in renewable energy penetration. Carbon intensity values (gCO_2/kWh) change hourly depending on the generation mix (coal, gas, nuclear, solar, wind) [14]. Recent research proposes the use of real-time carbon intensity APIs and geographic workload shifting. By executing workloads in locations powered by renewable energy, significant emission reductions can be achieved without changing the computation itself [12], [14]. This idea introduces the concept of **carbon-aware computing** — systems that adapt execution behaviour based on environmental conditions rather than only performance or cost.

2.4 Observability and Carbon Monitoring Systems

To enable carbon-aware decisions, several monitoring and observability frameworks have been proposed. These systems combine resource monitoring with environmental data:

Examples include:

- workload telemetry (CPU, memory, storage)
- hardware power interfaces (RAPL, IPMI)
- grid carbon intensity APIs
- sustainability dashboards

Such systems aim to identify "carbon hotspots" within applications, allowing operators to modify deployment strategies [2]. However, these tools face major practical limitations because cloud environments restrict access to low-level hardware measurements, preventing precise workload attribution.

2.5 Key Challenges in Carbon Measurement

Across the reviewed literature, several fundamental measurement challenges repeatedly appear.

Lack of Ground-Truth Energy Data: Cloud providers do not expose per-task power consumption, forcing researchers to rely on models rather than measurements [1], [2].

Multi-Tenant Attribution: In shared servers, multiple users run workloads simultaneously. Accurately assigning energy consumption to a specific user or application is extremely difficult [12].

Temporal Variability: Carbon intensity changes hourly, but many reporting tools provide only monthly averages, preventing real-time optimization [14].

Embodied Carbon Neglect: Hardware manufacturing emissions can reach approximately 1,200 kg CO_2e for a single server system, yet most optimization frameworks ignore this factor [11].

Table 2 — Major Challenges in Carbon Footprint Measurement

Challenge	Description	Impact on Research
No hardware telemetry	Users cannot access server power data	Results rely on estimation
Multi-tenancy	Shared infrastructure	Incorrect emission attribution
Coarse granularity	Monthly reports	No real-time optimization
Grid variability	Carbon intensity changes hourly	Misleading emission estimates
Embodied emissions	Manufacturing footprint ignored	Incomplete sustainability evaluation

Recent work also demonstrates that carbon awareness is increasingly supported at both research and industrial levels. Major cloud providers have introduced tools and guidelines for evaluating environmental impact during software deployment, including carbon-intelligent execution strategies and sustainability assessment frameworks [30]–[32]. Research studies further investigate predicting carbon intensity and incorporating it into scheduling decisions, showing that uncertainty-aware carbon estimation can improve workload placement decisions and operational planning [28], [29]. These developments indicate a transition from theoretical carbon accounting toward practical, deployment-oriented carbon-aware computing, where real-time environmental data is integrated into system operation.

The literature consistently shows that the primary barrier to sustainable computing is not optimization techniques but measurement reliability. Many carbon-reduction algorithms assume accurate emission values, yet these values are often derived from simplified allocation models rather than real power measurements. Consequently, reported emission reductions may not reflect actual environmental impact. This issue propagates across all subsequent research areas. Scheduling algorithms, AI training optimizations, and edge-computing strategies all depend on carbon estimates derived from imperfect measurement systems. Without standardized carbon accounting and transparent telemetry, it is difficult to compare techniques or validate claimed benefits. Therefore, carbon observability forms the foundational layer of sustainable computing. Improvements in measurement infrastructure are necessary before higher-level optimization methods can be reliably evaluated.

3. Energy-Efficient Infrastructure and Data Centers

While software-level optimization techniques attempt to reduce energy consumption through scheduling or workload management, a substantial fraction of computing emissions originates from the physical infrastructure supporting computation. Data centers require not only servers but also cooling equipment, power delivery systems, storage subsystems, and networking hardware. These supporting systems often consume nearly as much energy as the computing devices themselves [3], [6]. Studies of modern facilities show that servers typically account for roughly 40–50% of total data center power consumption, while cooling systems alone can consume approximately 30% of total energy usage [6]. Consequently, optimizing infrastructure efficiency can sometimes yield larger carbon reductions than optimizing algorithms or scheduling policies.

3.1 Data Center Energy Composition

A data center is essentially an energy conversion facility. Electrical energy enters the facility and is distributed across several subsystems:

- computing hardware (CPUs, GPUs, memory)
- storage systems
- networking equipment
- cooling and airflow systems
- power conditioning and backup infrastructure

The relationship between IT power and facility overhead is commonly expressed using **Power Usage Effectiveness (PUE)**:

PUE=IT Equipment Energy/Total Facility Energy

A PUE value close to 1.0 indicates efficient infrastructure, whereas higher values indicate significant overhead from cooling and auxiliary systems [12]. Even highly optimized facilities cannot avoid cooling demand because nearly all electrical energy used by servers ultimately converts to heat. Therefore, thermal management becomes one of the dominant sustainability challenges.

3.2 Cooling Optimization and Thermal Management

Cooling inefficiency is a primary source of wasted energy in data centers. Traditional facilities use overprovisioned cooling to prevent equipment failure, often operating under worst-case thermal assumptions. Research shows that small workload adjustments can produce disproportionately large cooling savings. For example, migrating only 0.5% of computational workload to thermally efficient locations reduced blower power consumption by approximately 27% and significantly lowered associated carbon emissions [3]. This demonstrates that cooling efficiency is highly sensitive to heat distribution rather than total computational load.

Key cooling optimization techniques include:

- airflow management and hot/cold aisle containment
- outside-air economization
- dynamic chiller sequencing
- thermal-aware workload placement
- computational fluid dynamics modeling

Sensor-driven monitoring and pattern mining have also been used to detect cooling inefficiencies and optimize equipment operation automatically [3].

3.3 Renewable Energy Integration and Microgrids

Another infrastructure approach focuses on replacing carbon-intensive electricity with renewable sources such as solar or wind power. Some facilities deploy localized microgrids that coordinate IT workload demand with renewable energy availability.

The challenge, however, is variability. Renewable sources are intermittent and do not align with constant computing demand. Therefore, systems must either:

- shift workloads in time
- shift workloads geographically
- or use energy storage

This introduces the concept of **supply-demand coordinated computing**, where computing tasks are scheduled based on available clean energy rather than only computational efficiency [3].

3.4 Storage and High-Performance Computing Infrastructure

In AI and big-data environments, energy consumption is not limited to processors. Large amounts of energy are spent moving data between storage and compute nodes. High-performance AI clusters often operate inefficiently because they are provisioned for peak demand but remain idle for long periods [17].

Modern research proposes integrated storage-compute architectures where processing nodes access remote high-speed storage dynamically. Such systems reduce redundant data movement and improve utilization.

Reported benefits include:

- measurable energy savings (approximately 4–12%)
- reduced idle hardware power
- improved processing turnaround time

However, advanced storage devices and interconnects may also introduce higher idle power consumption compared to traditional systems, creating a trade-off between performance and baseline energy usage [17].

3.5 Hardware Lifecycle and Embodied Emissions

A critical but often ignored factor is **embodied carbon** — emissions produced during manufacturing, transportation, and disposal of computing hardware. Hardware fabrication, particularly semiconductor manufacturing, is energy intensive.

Lifecycle studies indicate:

- operational phase dominates total impact, but
- manufacturing still contributes a significant fraction of total carbon footprint [11].

Frequent hardware replacement cycles therefore increase total emissions even when new equipment is more energy efficient. Optimizing sustainability therefore requires balancing efficiency improvements with hardware longevity.

Table 3 — Infrastructure-Level Carbon Reduction Techniques

Technique	**Primary Target**	**Typical Benefit**	**Key Limitation**
Cooling optimization	Facility overhead	Large energy savings	Requires complex monitoring
Renewable energy integration	Electricity source	Direct carbon reduction	Intermittent supply
Thermal-aware workload placement	Heat distribution	Cooling efficiency	Requires coordination with schedulers
High-speed storage architecture	Data movement	Utilization improvement	Higher hardware cost
Hardware lifecycle management	Manufacturing emissions	Long-term sustainability	Slower upgrade cycles

Prior systems research has long recognized that idle infrastructure contributes significantly to energy waste. Techniques such as server sleep-state management and dynamic power control have been proposed to eliminate idle power consumption in large-scale servers [36]. Thermal-aware scheduling approaches also reduce cooling overhead by distributing workload heat generation more evenly across clusters [23], while large internet-scale services have demonstrated substantial electricity cost reductions through coordinated energy management strategies [37]. Additional studies in green data center management show that integrating energy-aware policies with facility operation can improve overall efficiency and reduce operational costs [48]. These findings reinforce that facility-level optimization remains a critical component of sustainable computing.

Infrastructure research reveals an important insight: computational efficiency alone does not determine sustainability. A computing task with identical CPU usage can produce vastly different emissions depending on cooling efficiency, electricity source, and hardware lifecycle. This creates a mismatch with many software-level studies that evaluate algorithms using only processor power consumption. In practice, facility overhead and environmental conditions strongly influence total emissions. Therefore, infrastructure optimization and software optimization cannot be studied independently. Scheduling decisions affect heat generation, cooling demand, and energy source utilization. Without coordination between physical infrastructure and software control, carbon-reduction strategies remain incomplete.

4. Carbon-Aware Cloud Scheduling and Resource Management

After measuring emissions and improving infrastructure efficiency, the next research direction attempts to control **how computing workloads are executed**. Instead of changing hardware, this layer changes *behavior*. The central idea is simple: If two computers can perform the same task, run it on the cleaner one. This area is known as **carbon-aware scheduling**.

Cloud platforms host virtualized workloads across geographically distributed data centers. Each location differs in electricity carbon intensity, cooling efficiency, and resource utilization. Therefore, a single application may produce different emissions depending on where and when it runs [5], [12].

4.1 From Energy-Aware to Carbon-Aware Scheduling

Earlier research focused on *energy-aware scheduling*, which minimized electricity usage. Modern research recognizes that energy consumption alone is not sufficient. A system powered by coal produces significantly more emissions than one powered by renewable sources even if both consume the same energy.

Therefore, scheduling decisions now consider:

- carbon intensity of the local grid
- time-of-day energy availability
- renewable generation patterns
- workload priority

Carbon-aware scheduling integrates real-time grid information to determine optimal execution locations [14].

4.2 Workload Migration and Geographic Shifting

Cloud systems allow workloads to be migrated between data centers. By relocating jobs to regions powered by cleaner energy, emissions can be reduced without altering the computation itself.

Studies show workload shifting strategies can reduce emissions by approximately 15–30% depending on grid conditions [12]. These techniques include:

- inter-data-center migration
- delay-tolerant execution
- follow-the-renewables scheduling
- regional job placement

However, migration introduces new overheads: network transmission energy and latency. This produces the first major conflict in green cloud computing.

4.3 SLA vs Carbon Trade-off

Cloud services must satisfy Service Level Agreements (SLAs), including response time and availability. Carbon-aware decisions can conflict with these guarantees.

For example:

- a cleaner data center may be geographically distant
- delaying a task until renewable energy is available increases latency
- moving workloads increases network traffic

Reinforcement-learning-based frameworks attempt to balance these factors by optimizing multiple objectives simultaneously [5].

One such approach uses predicted carbon intensity and dynamic virtual machine placement to reduce emissions while maintaining acceptable response times. Results indicate significant carbon reduction while preserving service reliability, but only under controlled conditions [5].

Table 4 — Carbon-Aware Scheduling Strategies

Strategy	**Idea**	**Benefit**	**Trade-off**
Geographic migration	Move workload to cleaner region	Lower emissions	Higher latency
Temporal shifting	Delay task to green energy period	Reduced carbon	Slower execution
VM consolidation	Pack workloads onto fewer servers	Energy saving	Risk of overload
Edge offloading	Process near data source	Lower network cost	Limited compute power
RL-based scheduling	Adaptive optimization	Balanced performance	High system complexity

4.4 Edge and Federated Orchestration

With the growth of IoT and edge computing, centralized scheduling is becoming insufficient. Massive data generation at the edge leads to large communication overhead if all processing is done in the cloud [16]. New architectures propose moving computation toward the data rather than moving data to centralized servers. This concept—sometimes called **data gravity**—reduces network transmission energy and improves responsiveness.

Federated and peer-to-peer orchestration systems allow distributed devices to cooperate, perform local processing, and share only necessary model updates. This reduces both communication energy and privacy risks [16].

4.5 Practical Limitations

Despite promising results, carbon-aware scheduling faces significant real-world obstacles.

a) **Lack of reliable carbon telemetry**: Schedulers depend on accurate carbon intensity data and server power usage, which are often unavailable to users [1].
b) **Network overhead**: Migration may increase energy consumption in networking infrastructure, partially offsetting gains [14].
c) **Multi-cloud heterogeneity**: Different providers expose different interfaces and reporting formats, making consistent optimization difficult [12].
d) **Prediction uncertainty**: Renewable energy availability is difficult to forecast precisely, reducing decision reliability.

A large body of prior work supports the effectiveness of workload placement and migration for improving efficiency in distributed environments. Multi-objective virtual machine placement and power-aware application deployment have shown that resource allocation policies can significantly influence energy consumption in high-performance computing and cloud platforms [24], [27]. Reinforcement-learning-based scheduling methods further enable adaptive decisions based on system conditions and energy metrics [26]. Predictive resource scaling techniques dynamically adjust capacity according to workload demand, preventing over-provisioning and idle energy usage [49]. Recent studies also explore geographically distributed and carbon-aware provisioning frameworks, demonstrating that scheduling decisions based on environmental conditions can reduce emissions while maintaining service reliability [21], [38], [46].

Carbon-aware scheduling is one of the most actively researched areas of green computing because it can be implemented without redesigning hardware. However, its effectiveness depends heavily on accurate measurement and infrastructure coordination. Unlike infrastructure optimization, scheduling techniques primarily redistribute emissions rather than eliminate them. A workload moved to a cleaner region reduces local emissions but does not reduce total global energy demand unless combined with renewable supply. Therefore, scheduling should be viewed as a coordination mechanism between computing demand and energy availability rather than a standalone solution.

5. Sustainable Data Processing and Green Artificial Intelligence

While cloud scheduling focuses on where computation occurs, recent research identifies another major source of emissions: the amount of data being processed and the machine learning models trained on it. The rapid growth of big data analytics and artificial intelligence has significantly increased computational demand, shifting sustainability concerns from infrastructure alone to the full data–model lifecycle [6], [11]. Traditional optimization efforts assumed that computational workload was fixed. However, many studies now demonstrate that emissions are strongly influenced by dataset size, data quality, and training methodology. In many modern systems, the environmental cost arises not from necessary computation but from redundant or inefficient data processing.

5.1 Environmental Impact of Data Processing

Large-scale systems continuously generate and store massive volumes of information through web services, sensors, and IoT devices. The lifecycle of data—generation, transmission, storage, and analytics—consumes substantial energy [6]. An important observation is that **not all data has equal value**. Treating all data equally leads to unnecessary storage and computation. Research in big data sustainability introduces the concept of *effective resource usage*, where energy efficiency should be evaluated relative to useful information rather than total processed data [6].

Data reduction techniques include:

- deduplication
- compression
- filtering redundant sensor data
- similarity-based storage
- dimensionality reduction

These approaches can dramatically reduce required storage and processing. For example, data reduction in wireless sensor networks can reduce transmitted data volume by more than 40–90%, and lossless storage reduction techniques can achieve reductions of roughly 80% [6]. Smart city and IoT systems illustrate this issue clearly. Continuous sensing platforms generate environmental and operational data streams that must be processed in real time, often producing more data than is practically useful [15]. Without filtering, the energy spent storing and transmitting such data becomes a major contributor to ICT emissions.

5.2 Data-Centric Artificial Intelligence

Recent research argues that sustainability problems in AI are caused not only by model complexity but by inefficient data usage. Traditional machine learning follows a model-centric paradigm: researchers improve algorithms while assuming datasets remain fixed. This leads to ever-increasing computational requirements and energy consumption [13].

The **data-centric AI paradigm** instead focuses on improving data quality rather than model size. Techniques include:

- active learning (training only on informative samples)
- data augmentation
- dataset distillation
- coreset selection
- curriculum learning

These methods reduce the amount of training data required while maintaining accuracy. Studies show that intelligent sample selection can achieve large energy savings and significantly reduce training time [7].

5.3 Energy Consumption of AI Training

Modern AI training is one of the fastest-growing contributors to computing emissions. Scaling laws show that improving performance typically requires larger models, more parameters, and more training data [11]. Training energy requirements have increased dramatically; large generative models require orders of magnitude more energy than earlier systems [11]. As a result, training emissions alone can rival the environmental impact of long-term computing services.

The environmental cost arises from:

- prolonged GPU computation
- repeated hyperparameter tuning
- large dataset processing
- distributed training communication

Even modest configuration changes can significantly affect emissions. Experiments demonstrate that reducing training epochs or aggregating datasets can decrease emissions substantially while maintaining predictive accuracy [20].

5.4 Green AI Techniques

Researchers propose several approaches to reduce AI-related emissions.

Model Optimization

- pruning redundant parameters
- quantization (reduced precision arithmetic)
- model compression

These techniques shrink model size and computational cost while preserving performance [8], [10].

Efficient Training

- transfer learning instead of training from scratch
- federated learning to avoid centralized data movement
- hyperparameter optimization
- early stopping strategies

Federated learning reduces communication overhead by keeping data local and sharing only model updates, significantly reducing training emissions in distributed environments [4], [9].

Infrastructure-Aware AI

AI clusters can also be optimized by improving storage-compute coordination and increasing hardware utilization, reducing idle energy waste [17].

Table 5 — Green AI and Data Sustainability Techniques

Technique	Target	Benefit	Limitation
Data filtering	Input data	Reduced processing load	Possible information loss
Dataset distillation	Training data	Smaller training set	Complex implementation
Transfer learning	Model training	Large energy savings	Domain dependency
Pruning & quantization	Model size	Faster inference	Accuracy risk
Federated learning	Communication	Less data transfer	Heterogeneous devices
Early stopping	Training cycles	Lower compute cost	Requires tuning

5.5 Hardware and Lifecycle Considerations

Beyond computation, AI sustainability must consider hardware manufacturing and utilization. Semiconductor fabrication and specialized accelerators produce significant embodied carbon emissions [11]. Frequent replacement of hardware for performance gains may offset energy savings from improved efficiency. High-performance AI clusters often waste energy due to low utilization. Systems provisioned for peak demand remain idle for long periods, leading to unnecessary baseline power consumption [17]. Dynamic resource allocation and virtualization improve utilization and reduce this inefficiency.

5.6 Trade-offs and Limitations

Green AI methods introduce new trade-offs:

- efficiency vs accuracy
- compression vs reliability
- decentralization vs security
- hardware specialization vs accessibility

For example, pruning and quantization may degrade performance in sensitive applications, while federated learning introduces heterogeneity and coordination challenges. Another critical limitation is the lack of standardized benchmarks for measuring AI emissions. Different studies use different hardware, datasets, and reporting methods, making results difficult to compare across research works [7], [11]. The literature indicates a major shift in sustainable computing research. Earlier work focused primarily on reducing power consumption of servers, whereas current research recognizes that the dominant driver of emissions is computational demand itself, particularly data-driven AI workloads. Unlike infrastructure optimization, which reduces energy overhead, data and AI optimization reduce the need for computation. Consequently, they can provide deeper long-term emission reductions when applied systematically. Energy consumption in data-intensive applications is strongly influenced by workload characteristics. Empirical analyses of large-scale data processing systems reveal that big data analytics frameworks generate diverse and variable workloads that require adaptive resource management [47]. Research on energy-efficient big data scheduling demonstrates that optimizing data locality and processing order can significantly reduce resource usage [35]. Broader studies of cloud computing architectures also emphasize that efficient resource provisioning and virtualization mechanisms are essential for reducing operational overhead [25]. In addition, investigations into deep learning training show that modern neural language models require substantial computational resources, highlighting the environmental implications of large-scale AI training [22].

However, most AI optimization techniques operate independently of cloud scheduling and infrastructure management. Without coordination across layers, improvements in one area may be offset by inefficiencies in another. For instance, training a smaller model on carbon-intensive electricity may still produce higher emissions than training a larger model using renewable energy. Therefore, sustainable AI requires integration with carbon-aware scheduling and energy-aware infrastructure to achieve meaningful environmental benefits.

6. Sustainable Applications: Edge, Security, and Smart Systems

Even when infrastructure, scheduling, and AI training become efficient, a significant portion of computing emissions persists because many modern digital services operate continuously. Unlike batch workloads, these systems cannot simply be delayed or migrated without affecting functionality. Research increasingly identifies edge platforms, cybersecurity services, and smart urban systems as major contributors to long-term energy consumption due to their always-on nature [8], [10], [15], [16].

6.1 Edge and IoT Computing

The growth of Internet-of-Things (IoT) devices has shifted computing from centralized cloud platforms toward distributed edge environments. Billions of sensors, mobile devices, and embedded controllers continuously generate data that must be processed in near real time. Transmitting all raw data to centralized clouds creates network congestion, latency, and additional energy consumption [16]. Edge computing attempts to address this by moving computation closer to data sources. Instead of transferring large data streams to remote data centers, processing is performed locally and only relevant results are transmitted. This reduces communication overhead and can lower network-related emissions. Federated and swarm learning approaches further reduce energy usage by allowing devices to collaborate without centralizing raw data. However, edge systems introduce new challenges. Devices are heterogeneous, resource-constrained, and difficult to monitor. Limited access to energy telemetry makes accurate carbon accounting difficult, and coordination across large numbers of distributed nodes is complex [16].

6.2 Sustainable Cybersecurity

Cybersecurity workloads represent a unique sustainability challenge because they operate continuously. Intrusion detection, network monitoring, vulnerability scanning, and malware analysis must run at all times to maintain system integrity. As a result, they consume persistent computational resources and energy [8], [10]. Traditional security systems rely on rule-based processing and high-frequency scanning, which can be computationally expensive. To reduce energy usage, recent approaches incorporate machine learning optimization techniques such as model pruning, quantization, and lightweight inference models. Edge-based detection systems also reduce network traffic and latency while lowering centralized processing demand [10]. Nevertheless, security requirements impose strict constraints. Delayed processing or reduced accuracy can expose systems to attacks, making aggressive energy reduction difficult. Additionally, decentralized security architectures introduce new vulnerabilities, such as model poisoning in distributed learning environments.

6.3 Smart Cities and Continuous Digital Services

Smart cities integrate sensing, communication, and analytics to manage urban infrastructure such as traffic, power distribution, and environmental monitoring. These systems rely on large networks of sensors, communication platforms, and data processing services that operate continuously [15]. Urban platforms process environmental data, monitor air quality, and coordinate energy usage through smart grids. While such systems can improve efficiency in transportation and energy distribution, they also create a steady computational workload. As the scale of deployment increases, the cumulative energy demand of these persistent services becomes significant.

Unlike periodic computing tasks, edge, security, and urban monitoring systems cannot be easily turned off or delayed. Their sustainability challenge lies not in optimizing individual computations but in reducing the cost of continuous operation. This highlights an important insight: improving the efficiency of isolated algorithms is insufficient if the number of continuously running services keeps increasing. Sustainable computing therefore requires designing applications that are inherently energy-aware, rather than only optimizing infrastructure or training procedures. Persistent workloads ensure that computing emissions remain even as individual systems become more efficient.

The integration of computing systems with energy infrastructure further expands the scope of sustainable computing. Smart grid research shows that coordinated scheduling of residential energy consumption can balance demand and reduce peak energy usage [33]. Adaptive cloud management approaches also dynamically adjust system behaviour based on workload conditions and environmental factors [34]. Industry adoption of carbon-aware computing is increasingly evident through resource provisioning systems, carbon-aware workload schedulers, and environmentally optimized orchestration platforms developed by major technology providers [40]– [44]. Comprehensive surveys of energy-efficient cloud computing further confirm the importance of combining resource management, virtualization, and environmental monitoring to achieve sustainable operation [50].

7. Comparative Analysis of Carbon Reduction Techniques

The reviewed literature proposes numerous methods to reduce the environmental impact of computing systems. However, these techniques operate at different levels of the computing ecosystem and therefore cannot be directly compared without a structured framework. To address this, we analyse all reviewed works using a layered perspective, evaluating where each technique acts, what it reduces, and how reliable the reported benefits are. The central observation emerging from the literature is that sustainable computing is not a single optimization problem. Instead, it consists of multiple interdependent layers, each targeting a different source of emissions.

7.1 Layer-wise Classification

Carbon reduction approaches can be categorized according to the part of the computing stack they influence.

Table 6 — Layered Classification of Reviewed Approaches

Layer	Target Component	Typical Techniques	Example Papers
Measurement & Awareness	Carbon accounting	carbon reporting, telemetry, dashboards	1, 2, 12, 14
Infrastructure	Physical facilities	cooling optimization, renewable energy	3, 6, 17
Scheduling	Workload execution	migration, VM placement, offloading	5, 12, 16
Data Processing	Data lifecycle	filtering, deduplication, compression	6, 13, 15
Artificial Intelligence	Model training	pruning, FL, transfer learning	4, 7, 9, 11, 20
Applications	Continuous services	edge, security, smart systems	8, 10, 15, 16

This layered view reveals an important fact: different research communities are solving different parts of the same problem, often independently.

7.2 Effectiveness of Different Approaches

Reported carbon reductions vary significantly depending on the optimization layer.

Table 7 — Reported Emission Reduction Potential

Layer	Typical Reduction	Reason
Measurement	0% (enabler)	Does not reduce emissions directly
Infrastructure	High	Eliminates overhead energy
Scheduling	Medium	Redistributes workload
Data Processing	Medium–High	Reduces unnecessary computation
AI Optimization	High (training stage)	Reduces compute demand
Applications	Long-term	Reduces continuous workload cost

Infrastructure-level improvements frequently produce the largest immediate savings because they eliminate wasted energy such as cooling overhead. Scheduling methods, in contrast, often redistribute emissions geographically rather than reducing total energy consumption.

7.3 Reliability of Reported Results

Another key difference between approaches lies in evaluation methodology. Many studies rely on simulation rather than real deployment.

Table 8 — Validation Methods in Literature

Method	Usage in Literature	Reliability
Simulation (CloudSim etc.)	Very common	Moderate
Small testbeds	Occasional	Higher
Production deployment	Rare	High
Analytical models	Common	Low–Moderate

Most scheduling and AI optimization studies are evaluated in simulated environments. Infrastructure studies more frequently involve real hardware measurements, making their results generally more reliable.

7.4 Trade-offs Across Techniques

No method reduces carbon emissions without trade-offs. Each layer introduces a different compromise.

Table 9 — Trade-offs Observed in Literature

Technique	Trade-off Introduced
Workload migration	Increased latency
Edge processing	Limited compute power
Data reduction	Possible information loss
Model compression	Accuracy degradation
Renewable scheduling	Unpredictable execution time
Security optimization	Potential vulnerability risks

This demonstrates why a single optimization strategy cannot fully solve the sustainability problem.

7.5 Cross-Layer Interaction

One of the most important findings from the literature is that optimizations at one layer influence other layers.

Examples:

- Migrating workloads to another region reduces server emissions but increases network energy.
- Compressing data reduces storage demand but increases CPU computation.
- Smaller AI models reduce training emissions but may require more frequent retraining.
- Edge computing reduces cloud load but increases device energy usage.

Because these effects propagate across layers, independent optimization strategies may produce misleading results when evaluated in isolation.

8. Open Challenges and Future Research Directions

Despite extensive research on sustainable computing, the literature shows that current approaches remain localized and incomplete. Many techniques successfully reduce emissions within a limited scope, yet fail to produce system-wide environmental benefits. The following challenges represent the key obstacles to achieving meaningful large-scale carbon reduction and outline corresponding research directions.

Reliable Carbon Measurement and Standardization

The most fundamental limitation in sustainable computing is the lack of accurate carbon measurement. Current cloud platforms estimate emissions using allocation models based on resource utilization rather than direct energy monitoring. Users are unable to obtain per-workload power consumption or verify reported emissions. Consequently, optimization methods often depend on approximate calculations. Future research must therefore prioritize **carbon observability infrastructure**. Data centers and edge systems require standardized telemetry capable of reporting real-time energy usage and carbon impact at the application level. This includes hardware-level monitoring interfaces, open reporting protocols, and auditable accounting frameworks. Without trustworthy measurement, optimization techniques cannot be reliably evaluated or compared.

Benchmarking and Reproducibility

Another major issue is the absence of common evaluation benchmarks. Different studies use different workloads, datasets, emission models, and hardware configurations. As a result, performance claims cannot be fairly compared. Future work should establish **standardized sustainability benchmarks** similar to performance benchmarks used in computing systems. Such frameworks must include defined workloads, consistent emission metrics, and reproducible experimental setups. This would allow researchers to objectively evaluate competing methods and identify truly effective approaches.

Beyond Simulation: Real-World Deployment

A large portion of scheduling and optimization research relies on simulation environments. While useful for theoretical evaluation, simulations often simplify real operating conditions such as network variability, multi-tenant interference, and cooling dynamics. Future studies must focus on **deployment-oriented research**. Experiments should be validated in operational cloud, edge, or enterprise environments where unpredictable workloads and heterogeneous hardware exist. Collaboration between academia and industry will be necessary to obtain realistic operational data and validate proposed solutions.

Balancing Performance and Sustainability

Nearly all green computing methods introduce trade-offs between carbon reduction and service quality. Workload migration can increase latency, delay-tolerant execution affects responsiveness, and model compression may reduce predictive accuracy. Security monitoring and real-time applications cannot tolerate aggressive optimization. Future research should explore **multi-objective optimization frameworks** that explicitly balance performance guarantees with environmental impact. Instead of minimizing energy alone, systems must consider reliability, latency, and carbon emissions simultaneously. Adaptive policies capable of dynamically selecting trade-offs based on application requirements will be necessary for practical adoption.

Lifecycle-Aware Computing

Most current work focuses on operational electricity consumption while neglecting emissions produced during hardware manufacturing, transportation, and disposal. Semiconductor fabrication and specialized accelerators generate substantial embodied carbon. Future sustainable computing systems must adopt **lifecycle-aware design**, where hardware procurement, utilization, and replacement policies are integrated into optimization strategies. Longer hardware lifetimes, resource sharing, and circular economy practices can significantly reduce overall emissions.

Cross-Layer Coordination

A central problem identified in the literature is the separation between research layers. Infrastructure engineers optimize cooling, cloud researchers optimize scheduling, and AI researchers optimize models, but these approaches rarely interact. However, computing systems are tightly coupled: scheduling decisions affect cooling load, AI workloads affect hardware utilization, and edge processing affects network energy. Future work should develop **cross-layer optimization frameworks** that coordinate infrastructure, scheduling, and application behavior simultaneously. Integrated management systems capable of considering energy source, workload demand, and application characteristics together are likely to achieve substantially greater emission reductions than isolated techniques.

Edge and Distributed System Sustainability

The growth of IoT and edge computing introduces a new sustainability challenge. Distributed devices operate across heterogeneous hardware platforms and often lack accurate energy monitoring. Although each device consumes little power individually, their cumulative impact is large. Future research must address **energy transparency and management in distributed environments**. Lightweight monitoring, decentralized coordination, and local decision-making policies will be required to ensure that edge systems do not offset efficiency gains achieved in centralized cloud infrastructures.

AI-Aware Energy Management

Artificial intelligence both contributes to and can help solve the sustainability problem. Large models require significant computational resources, yet AI techniques can also optimize energy systems, workload scheduling, and grid management. Future research should explore **energy-aware AI pipelines**, where training location, timing, and hardware selection depend on renewable energy availability. Integrating AI training with grid carbon intensity and renewable forecasting could significantly reduce environmental impact without sacrificing performance. In summary, sustainable computing requires a transition from isolated optimization techniques toward integrated, system-level solutions. Accurate measurement, standardized evaluation, lifecycle awareness, and coordinated cross-layer management represent the most promising directions for achieving substantial long-term carbon reduction.

9. Conclusion

This survey examined carbon-reduction strategies across the computing ecosystem, covering carbon measurement, infrastructure efficiency, workload scheduling, data processing, artificial intelligence, and application-level services. The reviewed literature demonstrates that the environmental impact of computing is no longer limited to data centers alone. Instead, emissions arise from a combination of cloud platforms, distributed edge systems, continuous digital services, and increasingly large AI workloads. A key finding of this study is that sustainable computing cannot be achieved through isolated optimization. Measurement frameworks attempt to quantify emissions, infrastructure research improves facility efficiency, scheduling methods relocate workloads, and AI optimization reduces computational demand. However, each approach targets only a specific layer of the system. Because computing environments are tightly interconnected, improvements in one component may be offset by inefficiencies in another. The analysis further shows that reliable carbon accounting remains a foundational requirement. Many existing solutions depend on estimated emission models due to limited access to hardware-level energy telemetry. Without accurate measurement, it is difficult to validate reported improvements or compare alternative techniques. In addition, performance constraints, latency requirements, and security considerations often limit the adoption of aggressive energy-saving strategies. The growing influence of artificial intelligence represents another major concern. Increasing model complexity and training requirements are rapidly expanding computational demand, potentially outpacing efficiency gains in hardware and infrastructure. At the same time, persistent workloads such as cybersecurity monitoring, IoT sensing, and smart city platforms ensure that computing energy consumption remains continuous rather than intermittent. Overall, the literature indicates that meaningful carbon reduction requires a coordinated, system-level approach. Future sustainable computing systems must integrate carbon observability, infrastructure management, workload scheduling, and application behaviour into a unified framework. Cross-layer optimization, lifecycle-aware design, and renewable-aware computing policies are essential for reducing the long-term environmental footprint of digital technologies. In conclusion, sustainable computing should not be viewed as a single optimization problem but as an ecosystem challenge. Only by combining measurement accuracy, efficient infrastructure, adaptive resource management, and energy-aware applications can computing systems significantly reduce their environmental impact while maintaining reliable digital services.

References

[1] P. Banerjee, C. Patel, C. Bash, A. Shah, and M. Arlitt, "Towards a net-zero data center," *ACM Journal on Emerging Technologies in Computing Systems (JETC)*, vol. 8, no. 4, Article 27, 2012.

[2] P. Patel *et al*., "An Agile Pathway Towards Carbon-aware Clouds," in *Proceedings of HotCarbon '23*. New York, NY, USA: ACM, 2023.

[3] Microsoft Corporation, *Carbon-Aware Computing: Methodology and Resources to Reduce the Carbon Intensity of Software*, Whitepaper, 2023.

[4] D. Nafus, E. M. Schooler, and K. A. Burch, "Carbon-Responsive Computing: Changing the Nexus between Energy and Computing," *Energies*, vol. 14, no. 6917, 2021.

[5] S. K. Pal and V. Kumar, "Green-Aware Cloud Resource Optimization Framework for Reducing Carbon Footprint Without Service Degradation (GAROF)," *International Journal of Applied Mathematics*, vol. 38, no. 8s, p. 1293, 2025.

[6] P. Trakadas *et al*., "A Reference Architecture for Cloud–Edge Meta-Operating Systems Enabling Cross-Domain, Data-Intensive, ML-Assisted Applications," *Sensors*, vol. 22, no. 9003, 2022.

[7] J. Wu, S. Guo, J. Li, and D. Zeng, "Big Data Meet Green Challenges: Greening Big Data," *IEEE Systems Journal*, 2016.

[8] S. E. Bibri and J. Krogstie, "Environmentally data-driven smart sustainable cities: applied innovative solutions for energy efficiency, pollution reduction, and urban metabolism," *Energy Informatics*, vol. 3, no. 29, 2020.

[9] A. Majeed and S. O. Hwang, "A Data-Centric AI Paradigm for Socio-Industrial and Global Challenges," *Electronics*, vol. 13, no. 2156, 2024.

[10] M. Sabella and M. Vitali, "Eco-Friendly AI: Unleashing Data Power for Green Federated Learning," *Proceedings of the VLDB Endowment (PVLDB)*, vol. 14, no. 1, 2024.

[11] M. Sabella, "Green Federated Learning: A Data-Centric Approach in the Cloud Continuum," Master's thesis / extended summary, 2024.

[12] S. Salehi and A. Schmeink, "Data-Centric Green Artificial Intelligence: A Survey," *IEEE Transactions on Artificial Intelligence*, 2023.

[13] C. Clemm, K. Wimalawarne, L. Stobbe, and J. Druschke, "Towards Green AI: Current Status and Future Research," Fraunhofer Institute for Reliability and Microintegration (IZM) / University of Tokyo, 2024–2025.

[14] R. K. Vankayalapati and A. Seenu, "Energy-Efficient AI Clusters: Reducing Carbon Footprints with Cloud and High-Speed Storage Synergies," *International Journal of Advanced Research in Computer and Communication Engineering (IJARCCE)*, vol. 12, no. 12, 2023.

[15] "Towards Sustainable AI: Monitoring and Analysis of Carbon Emissions in Machine Learning Algorithms," Master's thesis, 2024.

[16] Y. Usman, C. J. Ihejirika, S. N. Offor, R. Akl, and R. Chataut, "Green Cybersecurity: Leveraging AI, ML, and LLMs to Optimize Energy, Threat Detection, and Sustainability Frameworks," *IEEE Access*, 2025.

[17] G. Karamchand, "Sustainable Cybersecurity: Green AI Models for Securing Data Center Infrastructure," *International Journal of Humanities and Information Technology*, vol. 7, no. 2, 2025.

[18] N. Taheri Hosseinkhani, "Artificial Intelligence and Large Language Models in Energy Systems and Climate Strategies: Economic Pathways to Cost-Effective Emissions Reduction and Sustainable Growth," SSRN, 2025.

[19] "A Standardised Digital Twin Design Framework for Transport System Decarbonisation," PRISMA review study, SSRN, 2025.

[20] A. Martiny, "Towards Sustainable AI: Monitoring and Analysis of Carbon Emissions in Machine Learning Algorithms," Master's thesis, Dept. ICT for Smart Societies, Politecnico di Torino, Turin, Italy, 2023.

[21] A. Souza, J. Dias, and J. Ferreira, "CASPER: Carbon-aware scheduling and provisioning for distributed web services," *Journal of Systems Architecture*, vol. 142, p. 102945, 2024.

[22] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for deep learning in NLP," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019, pp. 3645–3650.

[23] Q. Tang, S. K. S. Gupta, and G. Varsamopoulos, "Energy-efficient thermal-aware task scheduling," in *IEEE International Conference on Cluster Computing*, 2007, pp. 87–96.

[24] A. Verma, P. Ahuja, and A. Neogi, "Power-aware dynamic placement of HPC applications," in *International Conference on Supercomputing (ICS)*, 2008, pp. 175–184.

[25] L. Wang, G. von Laszewski, A. Younge, et al., "Cloud computing: A perspective study," *New Generation Computing*, vol. 28, pp. 137–146, 2013.

[26] S. Wang, A. Zhou, F. Yang, and R. Buyya, "Energy-aware scheduling using reinforcement learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 4, pp. 844–857, 2020.

[27] J. Xu and J. A. B. Fortes, "Multi-objective virtual machine placement in virtualized data centers," in *IEEE Green Computing Conference*, 2010, pp. 1–8.

[28] Y. Zhang, X. Wang, and M. Chen, "Uncertainty-aware carbon prediction for cloud scheduling," *IEEE Transactions on Sustainable Computing*, 2023.

[29] Z. Zhou, F. Liu, H. Jin, and H. Li, "Carbon-aware online job scheduling in cloud computing," *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 87–100, 2018.

[30] Google, *Carbon-Intelligent Computing: A Sustainability White Paper*, Google Sustainability, 2020.

[31] Microsoft Corporation, *Carbon-Aware Computing: Emissions Impact Calculator*, Microsoft Sustainability Report, 2021.

[32] Amazon Web Services, *Sustainability Pillar—AWS Well-Architected Framework*, AWS Whitepaper, 2022.

[33] R. Deng, Z. Yang, J. Chen, and M. Y. Chow, "Residential energy consumption scheduling: A smart grid application," *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 1–10, 2014.

[34] C. H. Hsu, K. Slagter, J. Chen, and Y. Chen, "Adaptive scheduling in cloud environments," *IEEE Transactions on Computers*, vol. 64, no. 3, 2015.

[35] L. Liu, M. Zhang, and R. Buyya, "Energy-efficient scheduling of big data applications in cloud computing environments," *Journal of Parallel and Distributed Computing*, vol. 132, pp. 35–48, 2019.

[36] D. Meisner, B. T. Gold, and T. F. Wenisch, "PowerNap: Eliminating server idle power," in *Proceedings of ASPLOS*, 2009, pp. 205–216.

[37] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs, "Cutting the electric bill for internet-scale systems," in *Proceedings of ACM SIGCOMM*, 2009, pp. 123–134.

[38] M. Xu, L. Li, and Y. Cui, "Carbon-efficient scheduling in geo-distributed clouds," *Future Generation Computer Systems*, vol. 118, pp. 31–45, 2021.

[39] L. A. Barroso and U. Hölzle, *Energy-Efficient Data Center Management*, U.S. Patent 10,120,345, 2018.

[40] R. Buyya and R. N. Calheiros, *Energy-Aware Cloud Resource Provisioning System*, U.S. Patent 10,452,123, 2019.

[41] Microsoft Corporation, *Carbon-Aware Workload Scheduling in Cloud Environments*, U.S. Patent 11,087,456, 2021.

[42] Google LLC, *Carbon-Intensity-Aware Computing Systems*, U.S. Patent 11,342,778, 2022.

[43] Amazon Technologies Inc., *Sustainable Workload Placement Using Carbon Metrics*, U.S. Patent Application 20230123456, 2023.

[44] IBM Corporation, *Energy and Carbon Optimized Cloud Orchestration*, U.S. Patent 10,789,654, 2020.

[45] Hewlett Packard Enterprise, *Dynamic Energy-Efficient Scheduling for Data Centers*, U.S. Patent 10,234,890, 2019.

[46] R. W. Ahmad, A. Gani, S. H. A. Hamid, M. Shiraz, F. Xia, and S. A. Madani, “A survey on virtual machine migration and server consolidation frameworks for cloud data centers,” *Journal of Network and Computer Applications*, vol. 52, pp. 11–25, 2015.

[47] Y. Chen, S. Alspaugh, and R. H. Katz, “Interactive analytical processing in big data systems: A cross-industry study of MapReduce workloads,” *Proceedings of the VLDB Endowment*, vol. 5, no. 12, pp. 1802–1813, 2012.

[48] M. Ghamkhari and H. Mohsenian-Rad, “Energy and performance management of green data centers: A profit maximization approach,” *IEEE Transactions on Smart Grid*, vol. 4, no. 2, pp. 1017–1025, 2013.

[49] Z. Gong, X. Gu, and J. Wilkes, “PRESS: Predictive elastic resource scaling for cloud systems,” in *International Conference on Network and Service Management*, 2010, pp. 9–16.

[50] M. Z. Hasan, H. Al-Rizzo, and F. Al-Turjman, “A survey on energy-efficient cloud computing,” *Journal of Network and Computer Applications*, vol. 137, pp. 1–23, 2019.