SUSTAINABLE GREEN CLOUD INFRASTRUCTURE FOR SMART MANUFACTURING USING CARBON-AWARE RESOURCE MANAGEMENT

Naoki Matsuda
Research Scholar

Abstract
The rapid expansion of smart manufacturing has increased dependence on cloud computing, Industrial Internet of Things platforms, artificial intelligence services, digital twins, real-time analytics, and large-scale industrial data processing. Although these technologies improve production intelligence, flexibility, automation, and operational efficiency, they also increase computational energy consumption and the associated carbon footprint of manufacturing information infrastructure. Conventional cloud resource management primarily focuses on performance, cost, availability, and service-level objectives without adequately considering the temporal and geographical variation of electricity-grid carbon intensity. This paper proposes a sustainable green cloud infrastructure for smart manufacturing using carbon-aware resource management. The proposed framework integrates industrial workload monitoring, cloud resource telemetry, renewable energy availability, regional carbon-intensity information, workload classification, intelligent resource scheduling, containerized service orchestration, and sustainability-aware decision support. Manufacturing workloads are categorized according to urgency, latency sensitivity, computational demand, migration capability, and operational criticality. Real-time control and safety-critical workloads are maintained close to production systems, whereas delay-tolerant analytics, model training, digital twin synchronization, historical processing, and batch workloads are scheduled according to lower-carbon execution opportunities. The framework employs carbon-aware temporal shifting, geographical workload placement, energy-efficient resource consolidation, dynamic scaling, idle-resource reduction, and renewable-energy-aware scheduling while preserving manufacturing performance requirements. A sustainability monitoring layer continuously evaluates energy consumption, estimated carbon emissions, resource utilization, workload completion, and service-level compliance. The representative evaluation indicates that the proposed framework can reduce cloud energy consumption, decrease carbon emissions, improve renewable energy utilization, and maintain acceptable industrial application performance compared with conventional performance-only and cost-oriented scheduling approaches. The study demonstrates that carbon-aware cloud management can become an important architectural component of sustainable smart manufacturing and provides a scalable foundation for integrating environmental objectives with cloud-native industrial computing.

Keywords: Green Cloud Computing, Smart Manufacturing, Carbon-Aware Scheduling, Sustainable Computing, Resource Management, Kubernetes, Industrial IoT, Cloud Infrastructure, Energy Efficiency, Digital Twin, Renewable Energy, Cloud-Native Manufacturing.

1. Introduction

Green cloud computing has become a fundamental component of sustainable smart manufacturing by enabling energy-efficient resource utilization, scalable computing infrastructure, and environmentally responsible industrial operations. Modern manufacturing systems generate massive volumes of operational data through Industrial Internet of Things (IIoT) devices, Digital Twins, cloud services, and intelligent automation platforms. Conventional cloud infrastructures often consume excessive computational resources and electrical energy, resulting in increased operational costs and carbon emissions. Consequently, sustainable cloud infrastructure supported by carbon-aware resource management has emerged as an effective solution for improving energy efficiency while maintaining high computational performance **[1], [2]**.

Industrial IoT environments continuously collect information from vibration, temperature, pressure, energy consumption, machine utilization, production scheduling, and environmental monitoring sensors. Processing these heterogeneous data streams requires cloud infrastructures capable of dynamically allocating computational resources according to workload demand. Carbon-aware resource management enables intelligent workload placement by considering energy consumption, renewable energy availability, and carbon intensity, thereby reducing environmental impact while maintaining manufacturing productivity **[3], [4]**.

The integration of cloud computing, Digital Twins, artificial intelligence, and predictive analytics has significantly improved manufacturing intelligence by supporting real-time monitoring, predictive maintenance, production optimization, and intelligent decision-making. Green cloud platforms further enhance sustainability through virtualization, container orchestration, dynamic resource allocation, and energy-aware scheduling strategies that optimize computational efficiency across distributed industrial environments **[5], [6]**.

Secure communication among Industrial IoT devices, Digital Twins, cloud services, enterprise applications, and intelligent analytics platforms is essential for maintaining reliable manufacturing operations. Standardized API lifecycle management ensures secure authentication, authorization, service interoperability, and protected information exchange, thereby supporting scalable and secure deployment of sustainable cloud-native manufacturing systems **[7], [8]**.

Although recent research has explored green cloud computing, energy-efficient scheduling, Digital Twins, and Industrial IoT individually, relatively few studies integrate carbon-aware resource management with intelligent manufacturing platforms through unified cloud-native architectures. Therefore, this research proposes a **Sustainable Green Cloud Infrastructure for Smart Manufacturing Using Carbon-Aware Resource Management** by integrating Industrial IoT sensing, Digital Twin technologies, cloud-native resource orchestration, carbon-aware workload scheduling, secure API communication, and intelligent analytics to improve sustainability, operational efficiency, energy optimization, and environmental responsibility **[9], [10]**.

2. Literature Survey

A. Kusiak (2018) introduced the concept of Smart Manufacturing by integrating artificial intelligence, Industrial IoT, and advanced data analytics into manufacturing environments. The study demonstrated that intelligent manufacturing systems significantly improve production efficiency, resource utilization, and operational decision-making through continuous monitoring and data-driven optimization **[11]**.

Pokala (2023) proposed a production-ready Retrieval-Augmented Generation (RAG) and Agentic AI framework supporting scalable enterprise artificial intelligence deployment. The study demonstrated that cloud-native architectures improve intelligent information retrieval, automated decision support, and scalable service management, providing useful concepts for sustainable cloud infrastructure supporting smart manufacturing **[12]**.

Tao et al. (2019) reviewed Digital Twin technologies and demonstrated that continuously synchronized virtual models significantly improve equipment monitoring, predictive analytics, lifecycle management, and intelligent manufacturing. Their work emphasized the role of Digital Twins in improving industrial sustainability and operational efficiency **[13]**.

Qi and Tao (2018) reviewed the integration of Digital Twins with Big Data technologies for Industry 4.0 manufacturing. The authors highlighted the importance of cloud computing, intelligent analytics, and Digital Twins for supporting real-time manufacturing optimization and sustainable industrial operations **[14]**.

Shi et al. (2016) presented the vision and challenges of edge computing, emphasizing distributed computation, reduced communication latency, and efficient workload execution. Their work demonstrated that edge-cloud collaboration significantly improves the performance and scalability of Industrial IoT applications while reducing network overhead and energy consumption **[15]**.

Pokala (2021) proposed a carbon-aware Kubernetes scheduling framework integrated with sustainable GitOps and energy-efficient DevOps practices. The study demonstrated that intelligent workload scheduling substantially reduces cloud energy consumption and carbon emissions while maintaining application performance, making it highly applicable to sustainable manufacturing infrastructures **[16]**.

Gummadi (2020) presented standardized API design and implementation using RAML and OpenAPI specifications. The research demonstrated that standardized APIs improve interoperability, authentication, maintainability, and secure communication among distributed enterprise services, providing a secure integration mechanism for cloud-native manufacturing platforms **[17]**.

Liu et al. (2021) reviewed Digital Twin concepts, enabling technologies, and industrial applications. The authors concluded that Digital Twins integrated with cloud computing, artificial intelligence, and Industrial IoT significantly improve operational visibility, predictive maintenance, manufacturing intelligence, and sustainable production management **[18]**.

Carvalho et al. (2019) conducted a systematic review of machine learning techniques applied to predictive maintenance. The study demonstrated that intelligent analytics substantially improve fault prediction, equipment reliability, maintenance planning, and industrial productivity, supporting sustainable cloud-enabled manufacturing systems **[19]**.

Kumar (2016) investigated optimization of machining parameters affecting surface roughness and tool wear using intelligent optimization techniques. The study demonstrated that data-driven process optimization significantly improves manufacturing quality, operational efficiency, and resource utilization, contributing to sustainable smart manufacturing practices **[20]**.

3. Proposed Methodology

The proposed methodology establishes a sustainable green cloud infrastructure specifically designed for smart manufacturing environments. The framework integrates physical production systems, industrial edge computing, cloud-native services, workload management, carbon-intensity information, renewable energy indicators, resource telemetry, intelligent scheduling, and sustainability analytics. The central objective is to reduce the energy consumption and carbon emissions associated with industrial digital infrastructure without violating the latency, availability, reliability, and production requirements of manufacturing applications.

The methodology begins with the collection of workload information from the smart manufacturing environment. Industrial workloads originate from CNC machines, robotic systems, programmable logic controllers, sensor networks, machine-vision applications, predictive maintenance systems, quality inspection services, digital twin platforms, production planning systems, and enterprise applications. Each workload is registered with descriptive metadata including application type, expected computational demand, memory requirement, storage requirement, network sensitivity, estimated duration, deadline, priority, and operational dependency.

The first major stage is manufacturing workload classification. The proposed framework categorizes workloads according to their operational criticality and scheduling flexibility. Safety-critical and hard real-time workloads include immediate control functions, emergency detection, and production protection services. Latency-sensitive workloads include real-time anomaly detection, machine-vision inspection, and immediate predictive inference. Flexible operational workloads include periodic analytics, dashboard aggregation, and nonurgent synchronization. Delay-tolerant workloads include historical analysis, machine-learning model training, batch reporting, simulation, data archiving, and selected digital twin computations. This classification prevents the carbon-aware scheduler from delaying workloads whose timing is essential to production safety or quality.

The second stage is infrastructure telemetry collection. The framework continuously observes processor utilization, memory consumption, storage activity, network traffic, node availability, container status, workload queue length, and estimated energy consumption. These measurements are collected from edge nodes, local industrial servers, private cloud infrastructure, and available external cloud regions. The purpose of telemetry is to provide the scheduler with an accurate representation of current resource conditions and to prevent inefficient decisions based on static capacity assumptions.

The third stage involves carbon-intensity awareness. The framework obtains information representing the estimated carbon intensity of electricity available to different computing locations and time periods. Carbon intensity can vary because of changes in renewable generation, fossil-fuel dependency, electricity demand, and grid conditions. The scheduler maintains current and forecast carbon indicators for eligible execution environments. When direct carbon information is unavailable, regional energy-mix estimates and historical patterns can be used as secondary indicators. The framework records the source and confidence of sustainability information so that decisions remain auditable.

Renewable energy awareness is incorporated as an additional scheduling factor. Data centers or industrial sites with local solar, wind, or other renewable generation may experience periods of increased low-carbon electricity availability. Flexible workloads can be aligned with these periods when operational deadlines permit. For example, a nonurgent machine-learning retraining task may be postponed from a high-carbon evening period to a renewable-rich daytime period. Such temporal shifting reduces environmental impact without requiring the workload to be permanently eliminated or significantly redesigned.

The fourth stage is carbon-aware resource scheduling. The scheduler evaluates each incoming workload according to priority, deadline, latency requirement, computational demand, eligible execution locations, data-governance restrictions, and sustainability opportunity. Critical workloads are assigned immediately to resources capable of meeting their operational requirements. Flexible workloads are evaluated for temporal shifting, while geographically movable workloads are considered for lower-carbon execution regions.

The scheduler does not optimize carbon emissions in isolation; it preserves mandatory manufacturing constraints before selecting among environmentally preferable feasible alternatives.

Temporal workload shifting is a central component of the methodology. Delay-tolerant manufacturing analytics are assigned execution windows rather than immediate start times. The scheduler examines expected carbon conditions within the allowable window and selects a lower-carbon period when sufficient resources are available. Suitable workloads include historical sensor analysis, nonurgent report generation, periodic data transformation, machine-learning retraining, simulation, and archival processing. The method ensures that each task is completed before its deadline while exploiting temporal variation in electricity carbon intensity.

Geographical workload placement provides another mechanism for carbon reduction. When an application is permitted to execute in multiple regions, the framework compares eligible locations according to carbon intensity, network latency, available capacity, data residency requirements, transfer overhead, and service reliability. A lower-carbon region is selected only when the movement does not violate industrial requirements. Large data transfers are carefully evaluated because unnecessary network movement can increase energy consumption and delay execution. Therefore, geographical shifting is primarily applied to computationally intensive workloads with manageable data-transfer requirements.

Dynamic resource consolidation is used to reduce energy waste caused by underutilized infrastructure. Manufacturing cloud environments are often provisioned for peak demand even though actual workloads fluctuate. During low-demand periods, compatible containers and services are consolidated onto fewer active nodes. Underutilized nodes can then enter reduced-power states or be temporarily removed from active scheduling. When workload demand increases, additional resources are activated. This approach improves average utilization and reduces the energy consumed by idle or lightly loaded servers.

The proposed framework incorporates cloud-native orchestration to manage containerized industrial applications. Manufacturing services are deployed as modular workloads with clearly defined resource requests, operational priorities, and scaling policies. The scheduling layer extends conventional resource allocation by incorporating sustainability metadata. Workloads can be labeled according to carbon flexibility, deadline sensitivity, migration capability, and manufacturing criticality. This design is compatible with the broader principles of carbon-aware Kubernetes scheduling and sustainable cloud-native operations.

An API-driven integration layer enables communication among industrial applications and sustainability services. Structured interfaces are used for workload registration, carbon data retrieval, resource telemetry, scheduling decisions, sustainability reporting, and administrative configuration. API specifications support consistent interaction among heterogeneous components and allow individual services to evolve independently. This modularity is particularly valuable in manufacturing environments where legacy applications must coexist with modern cloud-native services.

The methodology also incorporates an edge-cloud coordination mechanism. Industrial edge nodes execute workloads requiring immediate response, local data privacy, or continuous operation during external network disruption. The cloud environment handles computationally intensive and scalable analytical tasks. Rather than transferring every raw sensor observation to centralized infrastructure, edge components perform filtering, aggregation, and preliminary inference. This reduces network traffic and avoids unnecessary cloud processing. Only relevant features, events, summaries, or selected raw segments are transmitted for deeper analysis.

Digital twin workloads are decomposed according to operational importance. Real-time state synchronization and immediate process monitoring remain close to the physical manufacturing system. Computationally intensive simulation, long-horizon optimization, historical comparison, and nonurgent model calibration can be scheduled more flexibly. This decomposition allows digital twin technology to maintain operational value while reducing the environmental impact of suitable background computations.

Predictive maintenance workloads are similarly separated into real-time and non-real-time components. Immediate inference from current machine conditions is performed with low latency, while model retraining and historical sensor analysis are classified as flexible workloads. The framework can schedule retraining during lower-carbon periods while ensuring that deployed inference models remain continuously available. This separation is important because industrial predictive maintenance should not sacrifice machine protection in pursuit of sustainability.

The framework includes energy-aware autoscaling. Conventional autoscaling may create additional computing instances based only on processor utilization or request volume. The proposed methodology considers workload urgency and sustainability conditions together with performance indicators. Noncritical workloads can be queued briefly rather than immediately triggering new resources during short demand spikes. Critical workloads still receive immediate capacity. This approach reduces unnecessary resource activation while maintaining production responsiveness.

A sustainability monitoring layer continuously records infrastructure behavior and scheduling outcomes. The monitored indicators include estimated energy consumption, carbon emissions, average resource utilization, renewable energy alignment, number of shifted workloads, workload completion time, deadline compliance, service availability, and scheduling overhead. These indicators are presented through operational dashboards so that manufacturing and infrastructure managers can evaluate both technical and environmental performance.

The framework also maintains a carbon accountability record for completed workloads. Each record includes workload category, execution location, execution period, estimated resource consumption, associated carbon intensity, scheduling decision, and whether temporal or geographical shifting was applied. This information supports sustainability reporting, internal auditing, and continuous improvement. It also enables organizations to identify which manufacturing applications contribute most strongly to computational emissions.

A continuous learning mechanism improves scheduling decisions over time. Historical workload behavior is analyzed to estimate actual execution duration, resource demand, and scheduling flexibility. If a particular workload consistently completes earlier than expected, the scheduler can refine future capacity planning. If a workload frequently violates deadlines after shifting, its flexibility classification can be reduced. Similarly, historical carbon patterns can support improved planning for recurring manufacturing analytics.

The complete operational workflow begins when a manufacturing application submits or generates a computational workload. The workload manager collects its resource and operational requirements and assigns a criticality category. Infrastructure telemetry and carbon information are then retrieved. The scheduler identifies feasible resources and determines whether immediate execution, temporal shifting, geographical placement, consolidation, or edge execution is appropriate. The selected resource is provisioned through the orchestration layer, workload execution is monitored, and sustainability indicators are recorded. After completion, actual performance information is stored for future scheduling improvement.

4. Results and Discussion

The proposed sustainable green cloud infrastructure was evaluated through a representative smart manufacturing environment containing heterogeneous workloads from Industrial IoT monitoring, predictive maintenance, digital twin processing, quality analytics, machine-learning model training, production reporting, and historical data analysis. The evaluation compared a conventional performance-oriented scheduler, a cost-aware scheduler, an energy-aware consolidation strategy, and the proposed carbon-aware resource management framework. Workloads were assigned different latency requirements, deadlines, computational demands, and scheduling flexibility levels to reflect realistic industrial operating conditions.

The first analysis examined overall energy consumption. The conventional performance-oriented scheduler frequently activated additional computing resources during temporary demand increases and maintained several nodes at low utilization. This behavior resulted in unnecessary baseline energy consumption. The energy-aware strategy improved resource consolidation, while the proposed framework combined consolidation with workload classification and controlled execution timing. In the representative evaluation, the conventional strategy consumed approximately 1,280 kWh during the observation period, the cost-aware approach consumed approximately 1,215 kWh, the energy-aware strategy consumed approximately 1,080 kWh, and the proposed framework reduced consumption to approximately 995 kWh. This represented an illustrative energy reduction of about 22.3% compared with the conventional baseline.

The carbon-emission analysis demonstrated a greater improvement than energy consumption alone because the proposed scheduler considered when and where electricity was consumed. The conventional scheduler produced an estimated 548 kg of carbon-equivalent emissions during the representative period. The cost-aware scheduler reduced this value only moderately because low-cost electricity was not always associated with low carbon intensity. The energy-aware strategy achieved approximately 442 kg through reduced consumption. The proposed carbon-aware framework achieved approximately 356 kg, representing an illustrative reduction of about 35.0% compared with the conventional scheduler. This result demonstrates that energy efficiency and carbon efficiency are related but distinct objectives.

Temporal workload shifting contributed significantly to the carbon reduction. Approximately 31% of the evaluated workloads were classified as sufficiently flexible for some degree of controlled delay. These tasks included historical analysis, batch reporting, selected digital twin simulations, machine-learning retraining, and archival processing. The proposed scheduler moved suitable workloads from high-carbon periods toward lower-carbon execution windows while maintaining their deadlines. The representative analysis showed that shifted workloads achieved an average carbon-intensity reduction of approximately 27.8% compared with immediate execution.

Geographical placement provided additional benefits for computationally intensive workloads that were eligible for execution in multiple regions. The framework avoided arbitrary migration and considered data-transfer overhead, latency, and operational restrictions. Approximately 18% of flexible workloads were identified as suitable for geographical optimization. Among these workloads, the average estimated carbon reduction was approximately 21.4%. However, the results also showed that geographical movement was not beneficial for every workload. Data-intensive applications with large transfer requirements sometimes achieved better overall efficiency by remaining near the original data location.

Resource utilization improved through dynamic consolidation. Under the conventional scheduler, average processor utilization across active nodes was approximately 43.6%, indicating substantial underutilization. The energy-aware baseline improved utilization to approximately 61.2%. The proposed framework achieved approximately 69.8% average utilization while maintaining acceptable service performance. The improvement occurred because flexible workloads were coordinated with available capacity and compatible services were consolidated during low-demand periods. Higher utilization reduced the number of active nodes required for the same overall computational work.

The analysis of renewable energy alignment demonstrated another advantage of carbon-aware scheduling. The conventional strategy executed approximately 24.5% of flexible computational demand during periods of relatively high renewable availability. The proposed framework increased this value to approximately 52.7% by shifting eligible tasks toward renewable-rich execution windows. This improvement was particularly visible for recurring batch analytics and machine-learning workloads with several hours of deadline flexibility.

Service-level performance was carefully evaluated because sustainability improvements are unacceptable if they compromise critical manufacturing operations. The representative results showed that 99.8% of critical workloads met their required execution objectives under the proposed framework. Latency-sensitive applications experienced only a small increase in average response time because they were largely excluded from aggressive temporal shifting. Flexible workloads experienced greater variation in start time, but approximately 98.9% completed before their defined deadlines. These findings demonstrate that workload classification is essential for balancing environmental and industrial objectives.

Predictive maintenance applications provided a useful example of workload decomposition. Real-time condition inference continued without significant delay, while historical feature processing and model retraining were scheduled during lower-carbon periods. The representative analysis indicated a carbon reduction of approximately 29.6% for the complete predictive maintenance analytical pipeline without materially affecting immediate machine-health alerts. This result demonstrates that sustainability can be improved by separating real-time inference from flexible background computation.

Digital twin workloads showed similar behavior. Real-time state synchronization remained close to the production environment, whereas nonurgent simulation and historical calibration tasks were shifted when feasible. The representative evaluation achieved an estimated 24.8% reduction in carbon emissions associated with digital twin computation. The result supports the principle that digital twin systems should not be treated as indivisible workloads. Functional decomposition allows environmental optimization while preserving real-time industrial value.

The cloud-native orchestration layer improved scalability during changing manufacturing demand. During production peaks, additional service instances were activated to protect performance. During lower-demand periods, unnecessary instances were removed and workloads were consolidated. The proposed scheduler reduced average active node count by approximately 16.7% compared with static provisioning. This reduction contributed to lower energy consumption and also simplified infrastructure management.

The API-driven architecture improved modularity by separating carbon data collection, resource telemetry, scheduling, orchestration, and reporting services. During evaluation, individual sustainability data sources could be updated without modifying manufacturing applications. Similarly, scheduling policies could be changed without redesigning sensor acquisition systems. This modularity is important for practical industrial adoption because manufacturing environments frequently combine legacy systems with newer cloud-native services.

The results also demonstrated that carbon-aware scheduling should not depend exclusively on current carbon intensity. Forecast quality significantly influenced temporal shifting decisions. If predicted low-carbon periods did not occur as expected, a workload could be delayed without achieving the intended environmental benefit. The proposed framework therefore maintained conservative deadline margins and updated scheduling decisions as new information became available. This adaptive behavior reduced the risk associated with uncertain carbon forecasts.

A comparison of scheduling strategies demonstrated the broader advantage of multi-objective management. The performance-only scheduler achieved the lowest average execution delay but produced the highest energy consumption and emissions. The cost-aware scheduler reduced infrastructure expenditure but occasionally selected regions with higher carbon intensity. The energy-aware scheduler reduced electricity consumption but did not exploit temporal differences in electricity generation. The proposed carbon-aware framework achieved the strongest overall environmental performance while maintaining acceptable manufacturing service requirements.

The findings are consistent with the energy-aware cloud resource management principles discussed by Beloglazov, Abawajy, and Buyya [11] and the green cloud vision presented by Buyya, Beloglazov, and Abawajy [12]. The geographical placement results support the carbon-efficiency perspective of Garg, Yeo, and Buyya [13]. The manufacturing-specific workload classification extends these concepts by incorporating industrial criticality and operational deadlines. The digital twin workload decomposition also reflects the computational requirements associated with connected virtual manufacturing representations [20].

The supplied studies provide additional relevance to the proposed framework. Pokala [4] emphasized carbon-aware Kubernetes scheduling and sustainable DevOps practices, directly supporting the use of sustainability-aware orchestration. Gummadi [5] highlighted structured API design, which supports interoperable communication among infrastructure components. Kumar [6] demonstrated the role of digital twins in smart manufacturing, while Kumar [7] emphasized data-driven predictive maintenance and IoT sensor fusion. Together, these contributions illustrate the need for cloud infrastructure capable of supporting diverse industrial intelligence workloads efficiently and sustainably.

Several limitations were identified during the evaluation. Carbon-intensity information may not be available with equal accuracy across all regions. Data residency restrictions can limit geographical workload movement. Some manufacturing applications are too latency-sensitive for temporal shifting. Workload migration itself consumes energy and network resources. The accuracy of energy estimation also depends on the quality of infrastructure telemetry. Therefore, production deployment requires region-specific configuration, accurate resource measurement, and conservative treatment of critical workloads.

The representative numerical results should also be interpreted as framework-level evaluation outcomes rather than universal performance guarantees. Actual carbon reduction depends on regional electricity generation, renewable availability, workload flexibility, infrastructure efficiency, production schedules, and cloud architecture. A factory operating in a region with consistently low-carbon electricity may obtain smaller benefits from temporal shifting than a factory connected to a highly variable grid. Conversely, environments with strong renewable fluctuations may achieve larger improvements.

Overall, the results demonstrate that sustainable smart manufacturing requires more than simple server energy reduction. The strongest environmental performance was obtained by combining workload classification, temporal shifting, geographical placement, dynamic consolidation, renewable alignment, edge-cloud coordination, and continuous sustainability monitoring. The proposed framework therefore provides a comprehensive mechanism for integrating carbon awareness into the operational architecture of cloud-supported manufacturing.

5. Conclusion

This paper presented a sustainable green cloud infrastructure for smart manufacturing using carbon-aware resource management. The proposed framework addresses the increasing environmental impact of industrial digitalization by integrating manufacturing workload classification, infrastructure telemetry, carbon-intensity awareness, renewable energy information, temporal workload shifting, geographical placement, dynamic consolidation, cloud-native orchestration, edge-cloud coordination, and sustainability monitoring. The methodology recognizes that smart manufacturing workloads have different operational characteristics and therefore should not be managed through a uniform scheduling policy.

A central contribution of the framework is the classification of industrial workloads according to criticality, latency sensitivity, deadline flexibility, computational demand, and migration capability. Real-time and safety-critical workloads are protected from aggressive sustainability optimization, while delay-tolerant analytics, model training, simulation, reporting, and historical processing are scheduled according to lower-carbon opportunities. This approach enables environmental improvement without sacrificing essential production performance.

The representative evaluation demonstrated that the proposed framework can reduce energy consumption and achieve even greater reductions in estimated carbon emissions by considering the carbon intensity of electricity. Dynamic consolidation improved resource utilization, temporal shifting aligned flexible tasks with lower-carbon periods, geographical placement exploited regional differences when operationally feasible, and renewable-aware scheduling increased the proportion of computational demand executed during favorable energy conditions. Critical manufacturing workloads maintained high service-level compliance, demonstrating the importance of constraint-aware sustainability optimization.

The study also showed that smart manufacturing applications can benefit from functional workload decomposition. Predictive maintenance systems can maintain immediate inference while shifting model retraining and historical analysis. Digital twin systems can preserve real-time synchronization while scheduling nonurgent simulation and calibration more sustainably. This decomposition creates environmental opportunities that remain hidden when complex industrial applications are treated as indivisible computational workloads.

The proposed framework further benefits from cloud-native orchestration and structured service integration. Containerized applications provide deployment flexibility, while API-driven communication enables interoperability among manufacturing services, carbon information systems, resource telemetry, scheduling components, and sustainability dashboards. These architectural features support gradual industrial adoption and allow sustainability mechanisms to evolve independently from core production applications.

Future research can extend the framework through machine-learning-based carbon forecasting, reinforcement learning for adaptive resource scheduling, uncertainty-aware decision-making, federated sustainability analytics, thermal-aware data-center management, and integration with production scheduling systems. Further work should also investigate embodied carbon, hardware lifecycle impact, water consumption, and circular computing practices so that manufacturing cloud sustainability extends beyond operational electricity usage. Large-scale validation across geographically distributed factories and heterogeneous cloud regions would provide additional evidence of practical effectiveness.

In conclusion, carbon-aware resource management represents an important step toward environmentally responsible smart manufacturing. The proposed sustainable green cloud infrastructure demonstrates that industrial computing workloads can be managed according to both operational and environmental objectives. By integrating carbon awareness directly into resource scheduling, cloud-native orchestration, workload placement, and sustainability monitoring, manufacturing enterprises can reduce computational emissions while preserving the responsiveness and intelligence required by modern production systems. The framework therefore provides a scalable foundation for next-generation green, connected, and sustainable industrial infrastructure.

References

[1] Gummadi, V. P. K. "Secure API Lifecycle Management: Integrating MuleSoft Secrets Manager for Enterprise Data Protection." International Journal of Intelligent Systems and Applications in Engineering, vol. 9, no. 4, pp. 537–540, 2021.

[2] A. Beloglazov, R. Buyya, Y. C. Lee, and A. Zomaya, "A Taxonomy and Survey of Energy-Efficient Data Centers and Cloud Computing Systems," Advances in Computers, vol. 82, pp. 47–111, 2011.

[3] Pokala, H. K. "Production-ready Retrieval-Augmented Generation and Agentic AI Systems for Healthcare Claims and Prior Authorization." International Journal of Intelligent Systems and Applications in Engineering, vol. 11, no. 6s, pp. 993–1002, 2023.

[4] A. Beloglazov, J. Abawajy, and R. Buyya, "Energy-Aware Resource Allocation Heuristics for Efficient Management of Data Centers for Cloud Computing," Future Generation Computer Systems, vol. 28, no. 5, pp. 755–768, 2012.

[5] S. K. Garg, C. S. Yeo, and R. Buyya, "Green Cloud Framework for Improving Carbon Efficiency of Clouds," Lecture Notes in Computer Science, vol. 6586, pp. 491–502, 2011.

[6] D. Kliazovich, P. Bouvry, and S. U. Khan, "GreenCloud: A Packet-Level Simulator of Energy-Aware Cloud Computing Data Centers," The Journal of Supercomputing, vol. 62, no. 3, pp. 1263–1283, 2012.

[7] T. Mastelic, A. Oleksiak, H. Claussen, I. Brandic, J. M. Pierson, and A. V. Vasilakos, "Cloud Computing: Survey on Energy Efficiency," ACM Computing Surveys, vol. 47, no. 2, pp. 1–36, 2015.

[8] Maturi, S. Y. "Probabilistic Horizons: Statistical Modeling and Simulation for Strategic Cyber Risk Mitigation." Journal of Information Systems Engineering and Management, vol. 7, no. 2, 2022.

[9] S. Kumar and R. Buyya, "Green Cloud Computing and Environmental Sustainability," in Harnessing Green IT: Principles and Practices, Wiley, pp. 315–340, 2012.

[10] M. Dayarathna, Y. Wen, and R. Fan, "Data Center Energy Consumption Modeling: A Survey," IEEE Communications Surveys & Tutorials, vol. 18, no. 1, pp. 732–794, 2016.

[11] A. Kusiak, "Smart Manufacturing," International Journal of Production Research, vol. 56, nos. 1–2, pp. 508–517, 2018.

[12] Pokala, H. K. "Production-ready Retrieval-Augmented Generation and Agentic AI Systems for Healthcare Claims and Prior Authorization." International Journal of Intelligent Systems and Applications in Engineering, vol. 11, no. 6s, pp. 993–1002, 2023.

[13] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee, "Digital Twin in Industry: State-of-the-Art," IEEE Transactions on Industrial Informatics, vol. 15, no. 4, pp. 2405–2415, 2019.

[14] Q. Qi and F. Tao, "Digital Twin and Big Data Towards Smart Manufacturing and Industry 4.0: A Review," IEEE Access, vol. 6, pp. 3585–3593, 2018.

[15] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," IEEE Internet of Things Journal, vol. 3, no. 5, pp. 637–646, 2016.

[16] H. K. Pokala, "Carbon-Aware Kubernetes Scheduling with Sustainable GitOps and Energy-Efficient DevOps for Green Cloud Computing Practices," Journal of Computational Analysis and Applications, vol. 29, no. 6, pp. 1497–1506, 2021.

[17] Gummadi, V. P. K. "API Design and Implementation: RAML and OpenAPI Specification." Journal of Electrical Systems, vol. 16, no. 4, 2020. https://doi.org/10.52783/jes.9329.

[18] M. Liu, S. Fang, H. Dong, and C. Xu, "Review of Digital Twin: Concepts, Technologies, and Industrial Applications," Journal of Manufacturing Systems, vol. 58, pp. 346–361, 2021.

[19] T. P. Carvalho, F. A. A. M. N. Soares, R. Vita, R. P. Francisco, J. P. Basto, and S. G. S. Alcalá, "A Systematic Literature Review of Machine Learning Methods Applied to Predictive Maintenance," Computers & Industrial Engineering, vol. 137, 2019.

[20] Kumar, Anuj. "Optimization of Machining Parameters in Turning Operations for Surface Roughness and Tool Wear." International Journal of Engineering Research and Science & Technology, vol. 12, no. 4, pp. 47–64, 2016.