JOURNAL OF INTERNATIONAL CRISIS AND RISK COMMUNICATION RESEARCH
ISSN: 2576-0017
2026, VOL 9, NO 1

Green Federated Learning: Quantifying The Energy-Accuracy Trade-Off In Decentralized Iot Networks

Sai Teja Battula

Independent Researcher, USA

Abstract
Artificial intelligence systems have become increasingly energy-intensive. Deep learning model training consumes substantial electrical power across the global computing infrastructure. The environmental impact of such computational demands raises serious sustainability concerns within the machine learning community. Federated learning enables decentralized model training across distributed edge devices. Local computations occur on individual nodes without centralizing raw data. Privacy preservation and reduced communication overhead represent primary advantages of federated architectures. However, aggregate energy consumption across millions of participating edge devices remains poorly characterized. The Green-FL protocol introduces energy-awareness into federated learning optimization objectives. Dynamic resource allocation mechanisms classify devices based on real-time power source availability. Active nodes connected to renewable energy receive priority for training tasks. Battery-powered devices enter standby states until sustainable power becomes available. Training schedules adapt to fluctuations in clean energy availability across device networks. Experimental evaluation demonstrates substantial energy consumption reduction without significant accuracy degradation. Convergence time increases moderately due to intermittent node availability during low-renewal periods. Carbon emission projections indicate meaningful environmental benefits at the deployment scale. The accuracy-per-watt optimization metric provides a quantifiable framework for sustainable machine learning development. Edge computing environments benefit particularly from such energy-conscious training protocols.

Keywords: Federated Learning, Sustainable Computing, Energy Efficiency, Edge Computing, Carbon-Aware Scheduling, Internet Of Things.

I. Introduction

The rapid advancement of artificial intelligence has introduced significant environmental concerns. Deep learning models now require substantial computational resources for training. Natural language processing models, in particular, demand extensive energy during development. Research has demonstrated that training a single large transformer model can emit carbon dioxide equivalent to the lifetime emissions of multiple automobiles [1]. The energy consumption associated with neural architecture search proves even more alarming. Such iterative optimization processes multiply the carbon footprint by orders of magnitude. These findings have prompted the research community to examine the sustainability of current machine learning practices [1].

The environmental cost extends beyond model training alone. Hyperparameter tuning, model selection, and experimental iteration all contribute to aggregate energy demands. Graphics processing units operate at high power states for extended durations during these processes. Data centers housing such hardware consume electricity at industrial scales. The carbon intensity of this electricity varies based on regional power generation sources. Locations reliant on fossil fuels produce substantially higher emissions per computation. This geographical variability introduces additional complexity into sustainability assessments [1].

Federated learning has emerged as an alternative paradigm for distributed model training. This approach enables machine learning on decentralized data without requiring centralized aggregation. Local devices perform training computations on their own datasets. Only model updates, rather than raw data, are transmitted to coordination servers. This architecture preserves data privacy while reducing communication overhead [2]. Applications span multiple domains, including healthcare, finance, and mobile services. Sensitive information remains on source devices throughout the training process. The framework supports collaboration among multiple parties without exposing proprietary data [2].

However, federated learning introduces distinct energy considerations. Edge devices participating in training cycles consume additional power for local computations. Smartphones, tablets, and IoT sensors possess limited battery capacity. Continuous model training accelerates battery depletion on such devices. The aggregate energy impact across millions of participating nodes remains poorly understood. Current federated learning protocols prioritize model convergence and communication efficiency. Energy consumption at the device level receives minimal attention in existing optimization objectives [2].

This gap motivates the development of energy-aware federated learning frameworks. The Green-FL protocol proposed herein addresses sustainability within decentralized training contexts. The framework incorporates real-time power source information into scheduling decisions. Devices connected to renewable energy or grid power receive priority for training tasks. Battery-powered nodes enter standby states until sustainable power becomes available. This dynamic allocation balances model development requirements with environmental responsibility.

The protocol reimagines the optimization objective for federated systems. Traditional approaches minimize prediction loss as the sole criterion. Green-FL introduces energy efficiency as an explicit constraint within this formulation. Training schedules adapt to fluctuations in renewable energy availability across the device network. The approach demonstrates that sustainability and model performance need not conflict. Computational workloads can align with clean energy supply patterns without sacrificing accuracy. This research contributes a practical framework for environmentally conscious artificial intelligence development in resource-constrained edge environments.

II. Background and Related Work

A. Energy Consumption in Distributed Learning

The Federated Averaging algorithm established foundational principles for decentralized model training. This approach enables deep network optimization across distributed datasets without centralized data collection. Local devices perform multiple epochs of stochastic gradient descent before transmitting updates. The central server aggregates these updates through weighted averaging based on local dataset sizes. This methodology significantly reduces communication rounds compared to naive distributed approaches [3]. The original formulation demonstrated effectiveness on image classification and language modeling tasks. Mobile device keyboards served as a primary application domain for this technique. Character-level and word-level prediction models achieved competitive performance through federated training [3].

Communication efficiency constitutes a primary design consideration in federated systems. Network bandwidth constraints limit the frequency of model synchronization events. Increasing local computation between communication rounds reduces aggregate data transmission requirements. The trade-off between computation and communication shapes practical deployment decisions. Devices with limited connectivity benefit from extended local training periods. However, extended local training on heterogeneous data introduces optimization challenges [3]. Model divergence can occur when local datasets exhibit non-identical distributions. The Federated Averaging algorithm addresses this through periodic global aggregation. Convergence guarantees depend on the degree of data heterogeneity across participating nodes [3].

Statistical heterogeneity presents fundamental challenges for distributed learning systems. Real-world federated deployments encounter highly non-uniform data distributions. Each device generates data reflecting unique user behavior patterns. This non-independent and identically distributed nature violates assumptions underlying standard optimization theory [4]. Model performance can degrade substantially under severe distribution skew. Personalization techniques have emerged to address device-specific adaptation requirements. Multi-task learning formulations treat each device as a related but distinct learning problem. These approaches balance global model utility with local customization needs [4].
Systems heterogeneity introduces additional complexity beyond statistical considerations. Participating devices exhibit substantial variation in computational capabilities. Processing power, memory capacity, and storage availability differ across device populations. Network connectivity quality varies based on geographic and temporal factors [4]. Battery constraints limit sustained participation in training activities. Devices may become unavailable during training rounds due to power limitations. This intermittent availability complicates synchronization and convergence analysis. Stragglers with slow processing speeds can delay entire training rounds in synchronous protocols [4].
Privacy preservation motivates much federated learning research and deployment. Raw data remains on source devices throughout the training process. Only gradient updates or model parameters traverse network boundaries. This architecture reduces the exposure of sensitive information to central authorities. However, gradient information can potentially reveal details about the underlying training data. Differential privacy mechanisms add noise to transmitted updates for enhanced protection. Secure aggregation protocols prevent the server from inspecting individual device contributions [4]. The intersection of privacy requirements with energy constraints remains underexplored. Energy-efficient protocols must maintain privacy guarantees while reducing computational overhead. This balance shapes the design space for sustainable federated learning frameworks.

Table 1. Comparison of Federated Learning Challenges and Characteristics [3, 4].

Aspect	Description	Impact on Energy
Communication Efficiency	Reduced data transmission through local computation	Lower network energy overhead
Statistical Heterogeneity	Non-IID data distributions across devices	Variable computational requirements
Systems Heterogeneity	Diverse device capabilities and connectivity	Uneven power consumption patterns
Privacy Preservation	Gradient updates instead of raw data transmission	Additional encryption overhead
Synchronization	Periodic global model aggregation	Idle waiting periods increase energy waste
Straggler Effect	Slow devices delay training rounds	Extended active power states

B. Sustainable Computing Paradigms

Carbon-aware computing represents an emerging approach to environmentally responsible computation. This paradigm schedules workloads based on the carbon intensity of available electricity. Grid carbon intensity fluctuates throughout the day based on generation resources. Renewable strength availability varies with climate situations and time. Solar technology peaks during midday hours at the same time as wind speeds follow extraordinary cycles. Flexible computing tasks can shift to periods of lower carbon intensity. Data centers possess substantial flexibility in scheduling batch processing workloads [5].
Temporal load shifting exploits variations in carbon intensity over time. Non-urgent computations defer to hours when cleaner energy dominates the grid. Real-time carbon intensity signals inform scheduling decisions at the workload level. Geographic load shifting provides an additional dimension of optimization. Different regions exhibit distinct carbon intensity profiles based on local generation mix.

Workloads can migrate to locations with abundant renewable energy availability [5]. This spatial flexibility requires a distributed infrastructure spanning multiple grid regions. Cloud computing platforms increasingly support such carbon-aware placement decisions. The combination of temporal and spatial shifting maximizes emissions reduction potential [5].

Demand response integration connects computing infrastructure with grid management systems. Data centers can modulate power consumption based on grid operator signals. Peak demand periods often coincide with higher carbon intensity generation. Reducing computational load during these periods benefits both grid stability and emissions. Predictive models forecast carbon intensity to enable proactive scheduling decisions. Machine learning techniques improve the accuracy of these forecasts over time [5].

The integration of carbon awareness into computing infrastructure represents a fundamental shift. Traditional optimization focused exclusively on performance and cost metrics. Environmental impact now enters the objective function as an explicit consideration [5].

Quantifying carbon emissions from machine learning remains methodologically challenging. Multiple factors influence the environmental footprint of model training. Hardware selection affects energy consumption per computation performed. Training duration determines total energy requirements for model development. Geographic location determines the carbon intensity of consumed electricity [6]. Different regions exhibit order-of-magnitude variations in grid emissions factors. Models trained in coal-dependent regions produce substantially higher emissions than those trained with renewable power [6].

Standardized measurement frameworks enable meaningful comparison across studies. Reporting energy consumption alongside model performance metrics promotes transparency. The machine learning research community has begun adopting such reporting practices. Carbon calculators estimate emissions based on hardware specifications and training duration [6]. These tools raise awareness of the environmental costs associated with experimentation. Researchers can make informed decisions about computational resource allocation. The trade-off between model improvement and environmental impact becomes explicit [6].

However, existing carbon measurement approaches focus primarily on centralized training scenarios. Distributed and federated learning contexts introduce additional complexity. Edge device energy consumption occurs across heterogeneous hardware platforms. Power source information at the device level remains largely unavailable. Standard carbon intensity data applies to grid-connected infrastructure. Battery-powered mobile devices present distinct measurement challenges. Bridging this gap requires novel approaches to energy tracking in federated systems. The Green-FL protocol addresses this limitation through device-level power state monitoring.

Table 2. Carbon-Aware Computing Strategies and Mechanisms [5, 6].

Strategy	Mechanism	Application Domain
Temporal Load Shifting	Defer computation to low-carbon intensity periods	Batch processing workloads
Geographic Load Shifting	Migrate workloads to regions with renewable energy	Cloud computing platforms
Demand Response Integration	Modulate power based on grid operator signals	Data center operations
Carbon Intensity Forecasting	Predict renewable availability for scheduling	Proactive workload placement
Hardware Selection Optimization	Choose energy-efficient processing units	Model training infrastructure
Emissions Reporting Standards	Track and report carbon footprint metrics	Research transparency

III. Proposed Green-FL Protocol

A. Mathematical Formulation

Standard Federated Learning aims to minimize a global loss function F(w), which is a weighted average of local loss functions Fk(w) across K devices. The standard objective is defined as:

$$\min_w F(w) = \sum_{k=1}^{K} \frac{n_k}{n} F_k(w)$$

Where nk is the number of samples on device k, and n is the total sample count.

Green-FL modifies this objective by introducing an Energy Constraint. We define the energy cost Ek(t) for device k at training round t as the sum of computational energy (Ecomp) and communication energy (Ecomm):

$$E_k^t = E_{comp}(k,t) + E_{comm}(k,t)$$

The Green-FL optimization problem seeks to minimize the global loss subject to a sustainability constraint S:

$$\min_w F(w)$$

$$\text{s.t.} \sum_{k \in S_t} \mathbb{I}(Source_k = Green) \cdot E_k^t \leq \mathcal{B}_{carbon}$$

Where St is the subset of selected clients, I is an indicator function checking if the power source is renewable ("Green"), and Bcarbon is the allowable carbon budget for the training round.

B. Dynamic Resource Allocation Architecture

The Green-FL protocol establishes a hierarchical coordination framework for energy-aware federated learning. Edge devices transmit power state metadata to central aggregation servers at regular intervals. This metadata includes the current power source type and available energy reserves. The aggregation server maintains a real-time registry of device availability. Scheduling decisions leverage this registry to optimize both model convergence and energy sustainability.

Resource constraints fundamentally shape the design of edge computing systems. Devices at the network periphery possess limited computational capacity. Memory restrictions bound the complexity of local model architectures. Communique bandwidth varies primarily based on community situations and tool connectivity [7]. Adaptive algorithms need to account for this heterogeneity in aid availability. Control mechanisms dynamically adjust training parameters based on observed constraints. The frequency of local updates balances computation costs against communication overhead [7].

The Green-FL allocation algorithm classifies participating devices into three operational tiers. Active training nodes maintain a connection to renewable energy sources or stable grid power. These devices receive priority assignment for computationally intensive training tasks. Standby nodes operate on battery power with sufficient charge reserves. Such devices remain available for lightweight coordination tasks but defer intensive computation. Dormant nodes fall below the minimum charge thresholds required for safe participation. The protocol excludes dormant devices from training rounds to preserve device longevity [7].

Device tier assignments update dynamically as power conditions change. A smartphone transitioning from battery to charger moves from standby to active status. Solar-powered IoT sensors shift tiers based on ambient light conditions. This fluid classification enables responsive adaptation to fluctuating energy availability. The hierarchical structure ensures training proceeds with available sustainable resources [7].

C. Energy-Aware Training Scheduling

Training round initiation depends on sufficient active node availability. The protocol establishes minimum thresholds for green-powered device participation. Rounds commence only when renewable energy capacity meets computational requirements. This constraint prioritizes environmental sustainability over training speed. Model development proceeds at the pace permitted by clean energy availability.

Resource allocation optimization addresses the joint problem of computation and communication. Wireless network conditions influence the efficiency of model update transmission. Channel quality varies across devices based on location and interference [8]. Optimal resource allocation considers both

transmission power and local computation effort. Convergence analysis guides the selection of training parameters under resource constraints [8].
The Green-FL scheduler dynamically adjusts batch sizes based on aggregate renewable capacity. Periods of abundant clean energy support larger batch computations across the device network. Limited renewable availability triggers reduced batch sizes to match sustainable capacity. Local epoch counts similarly adapt to energy conditions. Extended local training occurs when green-powered nodes can sustain prolonged computation [8].
This energy-aware scheduling ensures computational intensity correlates with sustainable power availability. Arbitrary scheduling parameters give way to environmentally responsive adaptation. The protocol decouples training progress from fixed temporal schedules. Model convergence emerges from the cumulative contribution of sustainably powered computation. Peak renewable generation periods drive accelerated training activity. Low-carbon electricity windows receive preferential utilization for intensive model updates [8].

Table 3. Dynamic Resource Allocation States in Green-FL Protocol [7, 8].

Device Tier	Power Source	Training Role	Scheduling Priority
Active	Renewable or grid power	Full training participation	High
Standby	Battery with sufficient charge	Lightweight coordination tasks	Medium
Dormant	Below minimum charge threshold	Excluded from training rounds	None

IV. Experimental Methodology and Results

A. Experimental Setup

The experimental evaluation employed a simulated distributed environment comprising edge computing nodes. Raspberry Pi devices served as the representative hardware platform for IoT applications. Heterogeneous power configurations reflected realistic deployment scenarios. Some nodes maintained continuous grid power connections. Others operated on battery power with varying charge levels. A subset received simulated renewable energy inputs following solar generation patterns.
The CIFAR-10 image classification dataset provided the training workload for evaluation. Data partitioning across nodes introduced realistic distribution characteristics. Non-identical data distributions across devices represent a fundamental challenge in federated settings. Standard federated learning algorithms assume independent and identically distributed local datasets [9]. Real-world deployments violate this assumption due to user-specific data generation patterns. Performance degradation occurs when local data distributions diverge significantly from the global distribution [9].
Data partitioning strategies in the experiments reflected practical heterogeneity conditions. Each simulated device received a subset of classes rather than uniform sampling. This approach created realistic non-IID conditions across the device population. The severity of distribution skew varied across experimental configurations. Baseline comparisons utilized standard Federated Averaging without energy awareness. The Green-FL protocol operated under identical data distribution conditions [9].

B. Energy Consumption Analysis

Energy measurement instrumentation tracked consumption at the device level throughout training. Power draw varied based on computational activity and device state. Active training periods exhibited elevated energy consumption compared to idle states. The Green-FL protocol reduced aggregate energy consumption substantially compared to baseline approaches. This reduction emerged from intelligent scheduling rather than reduced total computation.
The temporal distribution of training activity shifted under energy-aware scheduling. Computation concentrated during periods of renewable energy availability. Battery-powered devices contributed

minimally to intensive training rounds. Grid-connected nodes absorbed the majority of the computational workload. This redistribution maintained model quality while improving sustainability metrics.

C. Performance and Convergence Analysis

Model accuracy evaluation compared Green-FL against standard federated averaging baselines. Final model performance remained comparable across both approaches. The accuracy differential fell within acceptable tolerance bounds for practical applications. Energy-aware constraints did not substantially compromise predictive capability.

Convergence time increased under the Green-FL protocol due to scheduling constraints. Reduced node availability during low-renewable periods extended training duration. The trade-off between convergence speed and energy sustainability favors environmental responsibility. Many practical applications tolerate extended training times for a reduced carbon footprint.

D. Infrastructure Heterogeneity Considerations

Large-scale computing infrastructure exhibits substantial performance heterogeneity. Nominally identical hardware demonstrates measurable variation in actual performance characteristics. Manufacturing differences and operational conditions contribute to this variation [10]. Warehouse-scale computing systems contain thousands of servers with differing capabilities. Workload placement decisions can exploit this heterogeneity for efficiency gains [10].

The Green-FL protocol leverages heterogeneity as an optimization opportunity. Device-level differences in power efficiency inform scheduling decisions. More efficient nodes receive preferential assignment during constrained periods. Carbon emission modeling incorporated regional electricity generation characteristics. Emissions calculations reflected the carbon intensity of consumed electricity. Projected savings at deployment scale demonstrated a meaningful environmental impact. The combination of energy reduction and carbon-aware scheduling multiplied sustainability benefits [10].

Table 4. Green-FL Simulation Environment Specifications [9, 10].

Component	Configuration	Purpose
Hardware Platform	Raspberry Pi nodes	Edge device representation
Dataset	CIFAR-10 partitioned	Image classification workload
Data Distribution	Non-IID across devices	Realistic heterogeneity simulation
Power Configurations	Grid, battery, and renewable	Heterogeneous energy sources
Baseline Comparison	Standard Federated Averaging	Performance benchmarking
Metrics Evaluated	Energy consumption, accuracy, and convergence time	Multi-objective assessment

Conclusion

The Green-FL protocol demonstrates the feasibility of integrating sustainability constraints into decentralized machine learning frameworks. Environmental responsibility need not compromise model performance in federated settings. Dynamic power-aware scheduling achieves substantial energy savings through algorithmic innovation rather than hardware modifications alone. Edge devices participating in training activities benefit from intelligent workload distribution based on power source characteristics. Renewable energy utilization receives prioritization through real-time device classification mechanisms. The three-tier allocation architecture effectively balances training requirements against sustainability objectives. Model convergence proceeds at the pace permitted by clean energy availability across the device network. Temporal flexibility inherent in many machine learning applications enables such environmentally conscious scheduling. The accuracy-per-watt metric offers practitioners a concrete framework for evaluating true computational costs. Future directions include integration with smart grid demand-response infrastructure for enhanced coordination. Carbon intensity forecasting could enable proactive scheduling decisions based on predicted renewable availability. Extension to larger and more computationally intensive model architectures warrants further attention. Geographic load shifting across distributed infrastructure presents additional optimization opportunities. The principles established through the Green-FL framework provide a foundation for responsible artificial intelligence deployment. Sustainable system mastering practices are becoming increasingly critical as computing proliferates across billions of connected devices globally.

References

[1] Emma Strubell et al., "Energy and Policy Considerations for Deep Learning in NLP," Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019. [Online]. Available: https://aclanthology.org/P19-1355.pdf
[2] QIANG YANG et al., "Federated machine learning: Concept and applications," arXiv, 2019.[Online]. Available: https://arxiv.org/pdf/1902.04885
[3] H. Brendan McMahan et al., "Communication-Efficient Learning of Deep Networks from Decentralized Data," Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, 2017. [Online]. Available: https://proceedings.mlr.press/v54/mcmahan17a/mcmahan17a.pdf
[4] Tian Li et al., "Federated Learning: Challenges, Methods, and Future Directions," arXiv, 2019. [Online]. Available: https://arxiv.org/pdf/1908.07873
[5] Ana Radovanovic et al., "Carbon-Aware Computing for Datacenters," IEEE TRANSACTIONS ON POWER SYSTEMS, 2023. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9770383
[6] Alexandre Lacoste et al., "Quantifying the Carbon Emissions of Machine Learning," arXiv, 2019. [Online]. Available: https://arxiv.org/pdf/1910.09700
[7] Shiqiang Wang et al., "Adaptive Federated Learning in Resource Constrained Edge Computing Systems," arXiv, 2019. [Online]. Available: https://arxiv.org/pdf/1804.05271
[8] Canh T. Dinh et al., "Federated Learning over Wireless Networks: Convergence Analysis and Resource Allocation," arXiv, 2021. [Online]. Available: https://arxiv.org/pdf/1910.13067
[9] Yue Zhao et al., "Federated Learning with Non-IID Data," arXiv, 2022. [Online]. Available: https://arxiv.org/pdf/1806.00582
[10] Jason Mars et al., "Heterogeneity in ‘Homogeneous' Warehouse-Scale Computers: A Performance Opportunity," 2011. [Online]. Available: https://www.researchgate.net/profile/Lingjia-Tang/publication/232634394