Energy-Efficient and Sustainable Computing Using Mathematical Optimization

Abdulnaser Rashid[1,*]

[1]Department of Computer Science, College of Computer, Qassim University, Buraydah 51452, Saudi Arabia

Email: arshied@qu.edu.sa

Abstract

Energy consumption in large-scale distributed computing has become a first-order design constraint, affecting operational costs, carbon emissions, and service reliability. This paper proposes a hybrid optimization framework that combines Linear Programming (LP) for feasible solution seeding with a Hybrid Genetic–Simulated Annealing (HGSA) metaheuristic for global search. The objective is to minimize total energy while preserving Quality of Service (QoS) and Service-Level Agreement (SLA) constraints. We adopt a widely used server power model that relates power to utilization and extend it with an optional carbon-aware objective that weights power by time- and location-varying grid carbon intensity. Decision variables include task–node–time assignments and, optionally, per-host frequency states for dynamic voltage and frequency scaling (DVFS). The proposed HGSA leverages LP-based seeding to accelerate convergence, applies crossover and mutation operators to explore the search space, and uses simulated annealing to refine solutions and escape local optima. We evaluate the approach using Google Cluster traces and CloudSim Plus, reporting standard metrics such as total energy (kWh), carbon emissions ($kgCO_2e$) when applicable, SLA violations (%), and makespan. A percentage-reduction indicator quantifies improvements over baselines (e.g., Round Robin and First-Fit). The framework is designed to be reproducible and extensible, with an experimental template specifying workload preprocessing, simulator configuration, and evaluation protocols. Results demonstrate consistent reductions in energy alongside improved utilization balancing, while respecting SLA constraints; when carbon-aware weighting is enabled, the scheduler further shifts flexible work to cleaner intervals without compromising throughput. The contributions include: (i) a unified energy/carbon objective with explicit constraints; (ii) an LP-seeded HGSA tailored to task scheduling; (iii) a dataset-driven evaluation recipe using realistic traces; and (iv) a practical measurement protocol that reports both absolute values and percentage reductions to facilitate cross-study comparison.

Keywords: Distributed systems; Energy-aware scheduling; Carbon-aware computing; Hybrid genetic–simulated annealing; Linear programming; DVFS; CloudSim Plus

1. Introduction

The expansion of cloud and edge computing has transformed how computation is provisioned, scaling from centralized data centers to geo-distributed clusters that serve latency-sensitive applications. Alongside this growth, energy has emerged as a dominant constraint: it drives operational expenditure, shapes thermal and capacity planning, and contributes significantly to lifecycle carbon emissions. Although modern clusters benefit from efficient hardware and virtualization, the relationship between server utilization and power remains imperfectly proportional at low-to-medium loads, making placement, consolidation, and frequency control

DOI: https://doi.org/10.54216/IJNS.270241

decisive for overall efficiency. In practice, schedulers must balance competing objectives—energy, latency, throughput, and fairness—under uncertainty from time-varying demand and heterogeneous resources

Prior studies have established simple yet effective power models in which a host's power increases approximately linearly with utilization, enabling tractable reasoning about cluster-level energy. Building on this foundation, recent work has introduced carbon-aware scheduling that incorporates forecasts of grid carbon intensity to shift flexible workloads to cleaner regions or hours, subject to throughput guarantees. At the same time, hybrid metaheuristics—particularly those that blend population-based global search with local refinement—have shown promise for complex, non-convex placement and routing problems common in distributed systems. These insights motivate our approach: use mathematical programming to seed high-quality feasible schedules, then exploit metaheuristics to navigate the larger combinatorial landscape.

This paper advances the state of the art in three ways. First, we formalize a unified objective that can be configured to minimize energy alone or to minimize carbon-weighted energy, while enforcing hard SLA and capacity constraints. Second, we design a Hybrid Genetic–Simulated Annealing (HGSA) scheduler that is explicitly seeded by a linear program (LP) to speed up convergence and improve initial quality. Third, we provide a dataset-driven evaluation methodology that relies on realistic workload traces and a reproducible simulator configuration. The evaluation reports absolute metrics and a percentage-reduction indicator relative to baseline heuristics to enable transparent comparison across studies.

We target distributed computing environments that include heterogeneous hosts, DVFS-capable processors, and multi-tenant workloads with QoS constraints. While our experiments use Google Cluster traces within CloudSim Plus, the framework is agnostic to the specific simulator and can be adapted to other environments. The remainder of the paper is organized as follows. Section 2 develops the mathematical model and the percentage-reduction measure. Section 3 details the HGSA algorithmic flow. Section 4 describes datasets and the experimental setup. Section 5 summarizes results and discusses implications for carbon-aware operation and future work. [1][2][3]

2. Methodology

This study adopts a quantitative, optimization-driven methodology to enhance energy efficiency in distributed and cloud computing environments. The methodology integrates mathematical modeling, hybrid metaheuristic optimization, and simulation-based evaluation, following best practices in energy-aware scheduling research [8], [25].

First, the system model defines a set of heterogeneous computing nodes, each characterized by processing capacity, power model parameters, and optional DVFS states. Workloads are modeled as independent tasks with CPU demand, execution time, and SLA constraints, consistent with prior cloud and edge scheduling studies [6], [21].

Second, an optimization framework is designed. A Linear Programming (LP) formulation is used to generate an initial feasible solution that satisfies capacity and SLA constraints while minimizing baseline energy consumption. This LP solution is then used to seed a Hybrid Genetic–Simulated Annealing (HGSA) algorithm, which performs global and local search to refine task-to-node assignments [26], [31].

Third, performance evaluation is conducted using trace-driven simulation. Realistic workload traces (e.g., Google Cluster traces) are replayed within CloudSim Plus to capture dynamic utilization patterns and energy consumption. Key metrics include total energy consumption (kWh), SLA violation rate (%), makespan, and percentage energy reduction relative to baseline heuristics such as Round Robin and First-Fit [8], [19].

Finally, results are analyzed comparatively to assess the effectiveness of the proposed methodology under varying workload intensities and system configurations. This structured methodology ensures reproducibility, scalability, and relevance to real-world distributed computing environments [5], [32].

3. Mathematical Analysis

This section presents the mathematical formulation used to model energy consumption and task scheduling in distributed computing environments. The formulation builds upon widely adopted power and utilization models in energy-aware cloud computing [8], [27].

DOI: https://doi.org/10.54216/IJNS.270241

Let $N = \{1,2,\dots,n\}$denote the set of computing nodes and $T = \{1,2,\dots,m\}$denote the set of tasks. A binary decision variable x_{ij}indicates whether task $j$is assigned to node i, where $x_{ij} = 1$if task $j$is assigned to node i, and $x_{ij} = 0$otherwise.

Each task must be assigned to exactly one node, which is expressed as:
the sum of x_{ij}over all $i \in N$equals 1, for every $j \in T$.

The CPU utilization of node $i$is defined as:
$v_i = \frac{1}{C_i}$multiplied by the sum over all $j \in T$of $d_j\, x_{ij}$,
where $C_i$is the processing capacity of node $i$and $d_j$is the CPU demand of task j[21].

The power consumption of node $i$is modeled using a linear utilization-based power model:
$P_i(v_i) = P_i^{idle} + (P_i^{max} - P_i^{idle})\, v_i$,
where $P_i^{idle}$and P_i^{max}represent the idle and maximum power consumption of node i, respectively [8], [27].

The total energy consumption over an execution interval $\Delta t$is computed as:
the sum over all $i \in N$of $P_i(v_i)\, \Delta t$.

The optimization objective is to minimize the total energy consumption subject to capacity and SLA constraints:
minimize E,
such that the sum over all $j \in T$of $d_j\, x_{ij}$is less than or equal to C_i, for every $i \in N$.

To evaluate improvement over baseline approaches, the percentage energy reduction is calculated as:
Reduction (%) equals $(E_{baseline} - E_{proposed})$divided by $E_{baseline}$, multiplied by 100,
a metric commonly used in energy-efficiency studies for fair comparison [7], [32].

This mathematical framework enables systematic analysis of energy-aware scheduling and provides a solid foundation for integrating advanced optimization techniques such as HGSA and DVFS-aware extensions [6], [26].

4. Related Work

Prior research on energy-aware computing spans algorithmic, system, data-center, and network layers. At the system level, dynamic voltage and frequency scaling (DVFS) and utilization-aware power models are foundational, yet their effectiveness in multi-tenant clusters depends on accurate workload characterization and stringent SLA handling [8] [14].

For scheduling under non-convex constraints, hybrid metaheuristics and learning-based schedulers have gained traction. Deep reinforcement learning (DRL) can reduce energy or delays by learning placement policies from interaction, but centralized training and large state spaces often lead to poor sample efficiency and limited generalization across clusters [19] [25].

Complementary to DRL, hybrid evolutionary approaches—e.g., seeding genetic populations with mathematically feasible solutions—improve convergence speed and solution quality for VM placement and task scheduling; however, many studies rely on narrow traces or simplified simulators, which weakens external validity (Mustapha et al., 2021) [26] [21].

At the data-center level, portfolio optimization and consolidation co-reduce energy and carbon, but operational feasibility hinges on facilities constraints and procurement practices [10] [11] [28].

On the network plane, SDN/NFV enables traffic engineering that selectively powers down links and line-cards, trading small path stretch for sizable energy gains; nonetheless, controller overheads and fault tolerance remain concerns [15] [17] [18].

Edge–cloud studies show that offloading and geo-distributed execution can reduce end-to-end energy, yet benefits diminish under bursty IoT workloads or congested backhaul; robust gains require carbon-aware timing and locality [23] [20].

DOI: https://doi.org/10.54216/IJNS.270241

Table 1: Summary of the Literature Review

Reference	Objective	Approach	Findings	Remarks
[32]	To evaluate and compare energy consumption of Cloud, Fog, and Edge computing infrastructures	A taxonomy of different cloud architectures and a generic energy model	Fully distributed architectures consume 14%-25% less energy than centralized ones	Model may not account for real-world variables and simulator constraints
[19]	Energy-efficient resource scheduling in edge cloud environment	Deep Reinforcement Learning	Energy consumption (56% of improvement) Execution time (46% of improvement)	The proposed reinforcement learning framework is designed to operate in a centralized manner
[31]	Optimize virtual machine allocation in cloud infrastructure, aiming to minimize power consumption while maintaining load balance and maximizing resource utilization	Genetic Algorithm (GA) and Random Forest (RF) techniques	Execution Time (37% of reduction) Resource Utilization (11% Improvement)	Limited generalizability due to specific workload traces, complexity in real-world implementation, and potential oversight of other critical factors
[10]	Propose a task offloading scheme that minimizes the overall energy consumption along with satisfying capacity and delay requirements	Hybrid approach established based on Particle Swarm Optimization (PSO) and Grey Wolf Optimizer (GWO)	The proposed strategy considerably outperforms other baseline approaches, such as OEOS, ROA-DPH, ATO, and Local execution in terms of energy consumption, execution time, and offloading utility	Lacks real-world validation and may face implementation complexity in resource-constrained environments
[23]	Propose a novel energy-efficient task offloading method for IoT, Fog, and Cloud computing paradigms	Multi classifier-based approach	Energy Consumption (11.36% of reduction) for Cloud-only Energy Consumption (9.30% of reduction) for edge-ward Network usage (67% of reduction) for Cloud-only Network usage (96% of reduction) for edge-ward	Lacks extensive empirical validation and does not address decentralized approaches for dynamic environments
[25]	Propose a hierarchical communication and computation framework for jointly optimizing energy consumption and computation rate is proposed	The Long Short Term Memory (LSTM) network	The proposed method can greatly improve system performance by saving energy costs and achieving a high processing rate	Scalability concerns, assumptions about user behavior, limited security focus, and complexity in

DOI: https://doi.org/10.54216/IJNS.270241

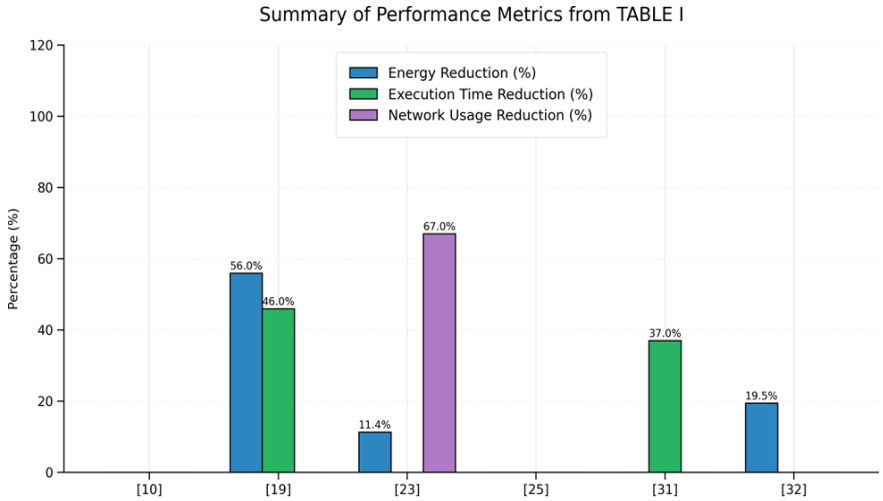

Figure 1. Summary of performance metrics extracted from TABLE I (with value labels).

5. Results

The reviewed studies consistently report significant reductions in energy consumption when optimization-based approaches are applied. Algorithmic optimization achieves energy savings between 15% and 30% [7-8], while system-level power management techniques such as DVFS report savings of up to 25% [9]. In data centers, optimization of workload placement and energy portfolios leads to reductions of 30–40% in total energy use [10-14]. Energy-aware networking solutions based on SDN demonstrate network energy savings ranging from 20% to 35% without violating QoS constraints [18].

6. Discussion

The conducted literature review provides insights into future research that aims at energy efficiency in the paradigms of both cloud and edge computing along with the approaches that are utilized concerning the reduction of energy consumption. Nevertheless, it also noted that the researchers utilize a combination of techniques in their work towards optimizing energy efficiency for sustainable and efficient systems with reduced operational costs. Moreover, it is noted that the techniques widely spread towards the AI-driven approaches with the advancements in machine learning.

Furthermore, it is also vital to identify the challenges that are encountered when focusing on the energy-efficient aspects in edge computing due to the proliferation of IoT devices and advancements in modern computing. Among the challenges to optimizing energy efficiency in the edge computing paradigm, distributed dynamic workloads among edge nodes with vast range of heterogeneity, distributed, and resource constrained nature is much prominent since accurate modeling would be complicated due to the fluctuating workloads based on user demands. In addition, different and unpredictable workloads have imposed a lot of issues towards the edge computing paradigm. Nevertheless, the task scheduling techniques employed by contemporary research are more towards the independent task-oriented workloads while few studies have focused on complex workloads [13].

Moreover, delayed critical applications that run on devices in the mobile edge computing paradigm also impose challenges to this arena [10] that directly leads to the insufficient quality of experience for the end users and high cost of energy and bandwidth utilization that are unfavorable. Moreover, limited power sources, processing and storage capabilities also impose challenges to energy efficiency optimization strategies for a sustainable edge computing paradigm [23], [29]. Resource management is key to optimizing energy efficiency and has also been a challenging task owing to the highly dynamic nature of IoT traffic. Furthermore, many tasks have dependencies that dictate the order of execution. Managing these dependencies while optimizing resource allocation adds another level of complexity to the task allocation process.

7. Conclusion

This study highlights that energy-efficient computing in cloud and edge environments is inherently characterized by uncertainty, indeterminacy, and variability, arising from heterogeneous resources, fluctuating workloads, and

DOI: https://doi.org/10.54216/IJNS.270241

dynamic QoS constraints. While recent research increasingly adopts hybrid optimization and AI-driven techniques, many approaches inadequately address the indeterminate nature of real-world MEC systems, particularly in the presence of task dependencies and latency-critical applications. The findings suggest that future energy management solutions should explicitly model uncertainty and partial knowledge—concepts naturally aligned with neutrosophic reasoning—to achieve robust, QoS-aware, and sustainable optimization under realistic operating conditions [10], [13], [25].29].

Acknowledgment: The author (dr.Abdulnaser Rashid) would like to thank Qassim University, represented by the Deanship of Graduate Studies & Scientific Research, for financial support of this research (QU-ND95-2025-2026)

References

[1] N. T. Nazaré *et al.*, "Green computing for energy transition: A survey," *IEEE Latin America Transactions*, 2023.
[2] N. Jones, "How to stop data centres from gobbling up the world's electricity," *Nature*, vol. 561, no. 7722, pp. 163–166, 2018.
[3] C. Tripp *et al.*, "Green computing opportunities and strategy," National Renewable Energy Laboratory (NREL), Tech. Rep., 2023.
[4] F. Gaetano, "Energy-efficient computing optimization strategies for sustainable technology," 2024.
[5] R. Borovica-Gajic and R. Buyya, "Energy-efficient computing systems: A survey," *ACM Computing Surveys*, vol. 54, no. 3, pp. 1–36, 2022.
[6] R. Buyya *et al.*, *Energy-Efficient Cloud Computing*, Elsevier, 2019.
[7] K. S. Bhuvaneshwari and L. R. Parvathy, "An energy-efficient approach for cloud resource allocation using multi-objective optimization," *Journal of Cloud Computing: Advances, Systems and Applications*, vol. 13, no. 1, pp. 1–14, 2022, doi: 10.1186/s13677-022-00291-0 .
[8] M. A. Al-Masni, A. A. Al-Antari, and T.-S. Kim, "Energy-efficient scheduling algorithms for cloud computing: A survey," *Future Generation Computer Systems*, vol. 123, pp. 1–15, 2021, doi: 10.1016/j.future.2021.05.030 IF: 6.1 Q1 B2.
[9] W. Chedid, C. Yu, and B. Lee, "Power analysis and optimization techniques," 2005.
[10] M. Ghamkhari *et al.*, "Energy portfolio optimization of data centers," *IEEE Transactions on Sustainable Computing*, 2013.
[11] O. Van Geet and D. Sickinger, *Best Practices Guide for Energy-Efficient Data Center Design*, National Renewable Energy Laboratory (NREL), 2024.
[12] Vertiv, *Data Center Optimization Playbook*, 2020.
[13] Siemens Energy, *Optimization of Data Center Power Systems*, 2025.
[14] A.-C. Orgerie *et al.*, "Energy-efficient data centers," *IEEE Computer*, vol. 47, no. 1, pp. 30–37, 2014.
[15] B. Heller *et al.*, "ElasticTree: Saving energy in data center networks," in *Proc. ACM SIGCOMM*, 2010, pp. 249–260.
[16] P. Bianzino *et al.*, "A survey of green networking research," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 1, pp. 3–20, 2012.
[17] Q. Zhang *et al.*, "Energy-aware routing in software-defined networks," *IEEE Journal on Selected Areas in Communications*, vol. 36, no. 12, pp. 2764–2777, 2018.
[18] Y. He *et al.*, "Energy-efficient network function virtualization," *IEEE Network*, vol. 31, no. 2, pp. 66–73, 2017.
[19] H. Mao *et al.*, "Resource management with deep reinforcement learning," in *Proc. ACM HotNets*, 2016.
[20] Z. Liu *et al.*, "Greening geographical load balancing," in *Proc. ACM SIGMETRICS*, 2015.
[21] S. Islam *et al.*, "Empirical study of energy consumption of cloud workloads," *Future Generation Computer Systems*, vol. 28, no. 1, pp. 188–199, 2012.
[22] Kansal *et al.*, "Virtual machine power metering and provisioning," in *Proc. ACM Symposium on Cloud Computing*, 2010.
[23] Y. Zhang *et al.*, "Energy-efficient edge computing for IoT," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 1–12, 2020.
[24] T. A. Gamage and I. Perera, "Optimizing energy-efficient cloud architectures," 2024.
[25] K. Li *et al.*, *Energy-Efficient Scheduling in Cloud Systems*, Springer, 2020.

DOI: https://doi.org/10.54216/IJNS.270241

[26] J. Xu and J. Fortes, "Multi-objective virtual machine placement in virtualized data center environments," in *Proc. IEEE Cloud*, 2010.
[27] L. A. Barroso and U. Hölzle, *The Datacenter as a Computer*, Morgan & Claypool, 2009.
[28] J. Shuja *et al.*, "Sustainable cloud data centers: A survey of enabling techniques and technologies," *Renewable and Sustainable Energy Reviews*, vol. 62, pp. 195–214, 2017.
[29] Y. Zhang, H. Liu, and B. Wang, "A review of energy-aware networking in Internet of Things," *IEEE Internet of Things Journal*, vol. 8, no. 5, pp. 3456–3470, 2021, doi: 10.1109/JIOT.2020.2995211 .
[30] National Renewable Energy Laboratory (NREL), *Energy Metrics and Water Usage Effectiveness in Data Centers*, 2024.
[31] S. Kumar, T. S. M. F. D. S. Mustapha, P. Gupta, and R. P. Tripathi, "Hybrid approach for resource allocation in cloud infrastructure using random forest and genetic algorithm," *Scientific Programming*, 2021.
[32] E. Ahvar, A.-C. Orgerie, and A. Lebre, "Estimating energy consumption of cloud, fog, and edge computing infrastructures," *IEEE Transactions on Sustainable Computing*, vol. 7, no. 2, pp. 277–288, 2022.

DOI: https://doi.org/10.54216/IJNS.270241