

A Multi-Resource Power Modeling Framework for Energy-Aware Cloud Simulation in CloudSim

Awet Teklemariam Ghebrekidan and Raju Shrestha 

Department of Computer Science, Oslo Metropolitan University (OsloMet)

Oslo, Norway

Email: {s362104; raju.shrestha}@oslomet.no

Abstract—The shift toward data-intensive and Artificial Intelligence (AI)-driven cloud workloads has rendered traditional Central Processing Unit (CPU)-centric power modeling increasingly insufficient for accurate datacenter energy estimation. While CloudSim is the de facto standard for evaluating energy-aware strategies, its native models predominantly focus on CPU utilization, effectively overlooking the non-trivial power contributions of auxiliary subsystems. This paper presents a modular multi-resource power modeling framework for CloudSim that integrates subsystem-level energy estimation into a unified simulation workflow. The framework introduces three critical extensions: (i) a high-resolution monitoring module capturing time-series utilization for CPU, Random Access Memory (RAM), bandwidth, and storage; (ii) modular, resource-specific power models that facilitate disaggregated energy attribution; and (iii) enhanced datacenter components supporting multi-dimensional resource capacities and tiered energy analytics. By providing a cohesive abstraction layer and unified Application Programming Interface (API), the framework enables consistent and extensible energy evaluation across heterogeneous infrastructure scales. Evaluation under high-intensity workloads indicates that traditional CPU-centric models obscure a non-trivial share of subsystem-level energy consumption, with non-CPU components representing approximately one-third of the total energy footprint in the evaluated stress-test scenario.

Keywords—Cloud computing; energy efficiency; power modeling; multi-resource simulation; CloudSim; datacenter sustainability.

I. INTRODUCTION

The digital transformation of the global economy, accelerated by Artificial Intelligence (AI)-driven workloads and data-intensive services, has necessitated a critical re-evaluation of datacenter energy efficiency. Recent projections estimate that datacenters accounted for approximately 1.5% of global electricity consumption in 2024, with a trajectory nearing 3% by 2030 [1]. Within these infrastructures, Information Technology (IT) equipment represents the primary energy sink [2]. While the Central Processing Unit (CPU) has historically been viewed as the dominant contributor, empirical studies increasingly show that memory, storage, and network subsystems account for a substantial and growing share of total system energy [3][4].

Despite this shift, current simulation-based research remains largely tethered to *CPU-centric power models*. These models estimate host power solely as a function of CPU utilization, implicitly assuming that auxiliary subsystems are either negligible or scale proportionally with computational load. Measurement-based studies contradict this assumption,

demonstrating that memory refresh operations, storage I/O, and network traffic exhibit power behaviors that are weakly correlated or entirely decoupled from CPU activity [5][6]. Consequently, relying on CPU-only models introduces a visibility gap that can lead to skewed energy evaluations and misleading conclusions when testing energy-aware cloud management policies [7].

CloudSim [8] is the most widely used simulation toolkit for cloud resource management research. However, its native power modeling capabilities focus primarily on CPU utilization and do not provide an integrated mechanism for aggregating subsystem-level energy consumption. Although several extensions exist that model individual resources such as networking or storage [9][10], these solutions are typically domain-specific and operate in isolation. As a result, researchers lack a unified and extensible framework for holistic host-level energy estimation within CloudSim.

To address this limitation, this paper proposes a modular multi-resource power modeling framework for CloudSim. The contribution of this work lies in the integration and unification of subsystem-level power models into a cohesive simulation framework, building upon existing empirical modeling approaches [5][6]. The framework enables fine-grained energy attribution across CPU, memory, storage, and network subsystems while preserving compatibility with existing CloudSim scheduling and allocation policies.

The key contributions of this work are:

- *A unified multi-resource simulation framework* that integrates CPU, memory, storage, and network power estimation within CloudSim.
- *Granular energy attribution* at the virtual machine (VM), host, and datacenter levels to improve interpretability and reproducibility of simulation studies.
- *Quantification of modeling bias*, demonstrating that CPU-only models structurally misestimate energy consumption under data-intensive workloads.

The remainder of this paper is organized as follows. Section II reviews related work on datacenter power modeling and CloudSim extensions. Section III introduces the CPU-centric baseline models used for comparison. Section IV presents the proposed multi-resource framework and its integration into CloudSim. Section V describes the experimental setup and workload design. Section VI presents and discusses the evaluation results. Finally, Section VII concludes the paper and outlines future research directions.

II. BACKGROUND AND RELATED WORK

This section reviews prior work on server power modeling and cloud simulation, highlighting the need for a unified multi-resource energy model.

A. Power Consumption Dynamics in Modern Servers

Modern cloud servers are composed of several power-consuming subsystems, including Central Processing Units (CPUs), memory, storage, and network interfaces. While CPUs have historically been the primary focus of energy modeling, recent empirical studies demonstrate that non-CPU components now contribute a substantial and growing portion of total energy consumption by the servers [3][4]. Memory subsystems consume significant power due to constant background refresh cycles, even during low utilization; storage devices incur energy costs during state transitions and Input/Output (I/O) operations; and network interfaces draw power relative to data transfer rates [5][6]. Crucially, these components exhibit power behaviors that are weakly correlated with CPU utilization, rendering traditional CPU-only models structurally incapable of reflecting true server power dynamics.

B. Cloud Simulation Frameworks and Extensions

CloudSim [8] is the most widely adopted toolkit for evaluating cloud resource management strategies, including allocation, scheduling, and energy-efficient consolidation. Its modular design has enabled several domain-specific extensions, such as CloudSim Plus [11] for structural improvements, NetworkCloudSim [9][12] for communication modeling, and CloudSimDisk [10] for disk I/O simulation. While these tools improve modeling fidelity within individual subsystems, they typically operate in isolation and do not provide an integrated host-level energy aggregation layer.

Other simulation frameworks, such as GreenCloud and iFogSim, focus on network-centric and edge computing scenarios, respectively. However, none of these frameworks offer a unified and extensible abstraction layer that aggregates CPU, memory, storage, and network power into a single host-level energy profile. As a result, most CloudSim-based energy-aware studies continue to rely on CPU-centric power models despite well-documented subsystem-level power diversity.

C. Single-Resource vs. Multi-Resource Energy Modeling

Energy-aware cloud research has traditionally emphasized CPU-oriented techniques, such as Dynamic Voltage and Frequency Scaling (DVFS), utilization-based scheduling, and virtual machine consolidation [13]. Evaluating these techniques through CPU-only power models implicitly assumes that non-CPU energy scales with computational load. Measurement-based studies contradict this assumption, demonstrating independent and often non-linear power behaviors across CPU, memory, and network subsystems [5][6].

Recent simulation efforts have begun to incorporate selected non-CPU components [14]. However, these approaches typically lack full subsystem coverage, unified aggregation, or modular extensibility, limiting their applicability for evaluating cross-resource energy-management strategies.

D. Research Gap and Motivation

Despite growing recognition of multi-resource energy behavior, existing simulation approaches lack a comprehensive and modular solution for host-level power modeling in CloudSim. Specifically, there is an absence of: (i) an integrated model encompassing CPU, memory, storage, and network power, (ii) a unified host-level energy aggregation layer, and (iii) fine-grained subsystem-level attribution. These limitations motivate the need for a simulation framework that coherently integrates existing subsystem models into a unified, extensible energy estimation workflow.

III. CPU-CENTRIC BASELINE MODELS

To establish a reference for comparison, we employ the two most commonly used CPU-centric power models in CloudSim-based energy-aware studies. These models are not incorrect per se, but they abstract host power consumption as a function of CPU utilization alone, implicitly aggregating all subsystem energy into a single dimension. They therefore serve as appropriate baselines for quantifying the modeling bias introduced by single-resource assumptions.

A. Linear CPU-Only Power Model

The linear model estimates host power by interpolating between idle and maximum CPU load, treating all non-CPU power as a fixed component of the idle state. This approach is widely adopted due to its simplicity and low computational overhead.

$$P_{\text{host}}^{\text{lin}}(t) = P_{\text{idle}} + (P_{100} - P_{\text{idle}}) U_{\text{cpu}}(t) \quad (1)$$

where P_{idle} and P_{100} represent the measured host power at 0% and 100% CPU utilization, respectively, and $U_{\text{cpu}}(t)$ denotes the instantaneous CPU utilization. By construction, this model assumes that memory, storage, and network subsystems scale proportionally with CPU load, an assumption that does not hold under data-intensive workloads.

B. SPECpower-Based CPU Model

The SPECpower-based model relies on empirical measurements obtained from the SPECpower_ssj2008 benchmark [15]. It maps discrete CPU utilization levels to measured power values and interpolates between them to obtain a continuous power curve.

$$P_{\text{host}}^{\text{spec}}(t) = \text{SPEC}(U_{\text{cpu}}(t)) \quad (2)$$

where $\text{SPEC}(\cdot)$ denotes the benchmark-derived lookup and interpolation function. Although this model captures non-linear CPU power behavior more accurately than the linear approach, it still encapsulates the entire host energy consumption within a single CPU-driven function.

Both baseline models are computationally efficient and widely accepted in the literature. However, neither provides visibility into the independent contributions of non-CPU subsystems, nor can they capture energy variations caused by memory-intensive, I/O-intensive, or network-heavy workloads. These limitations motivate the need for a multi-resource modeling approach.

IV. MULTI-RESOURCE POWER MODELING FRAMEWORK

To address the limitations of CPU-only host power abstraction, we extend CloudSim with a unified Multi-Resource (MR) power modeling framework that aggregates subsystem-level energy estimation into a single host-level profile. The objective of the framework is not to redefine physical power models, but to coherently integrate established subsystem-level models into a modular and extensible CloudSim-compatible architecture.

A. Framework Overview

The proposed framework extends CloudSim with a modular, multi-resource energy modeling layer designed to: (i) provide holistic host power estimation through subsystem-specific models, (ii) support heterogeneous hardware configurations, (iii) enable modular substitution or calibration of subsystem models, and (iv) support VM, host, and datacenter-level energy accounting.

As shown in Figure 1, the framework is implemented as a non-intrusive extension layer that interfaces with native CloudSim entities through well-defined APIs. A Monitoring Module periodically retrieves fine-grained utilization metrics from hosts and VMs, including Central Processing Unit (CPU) utilization, Random Access Memory (RAM) usage, storage I/O activity, and Network Interface Card (NIC) traffic. These metrics are processed by independent subsystem power models, each responsible for estimating instantaneous power consumption for its corresponding hardware component.

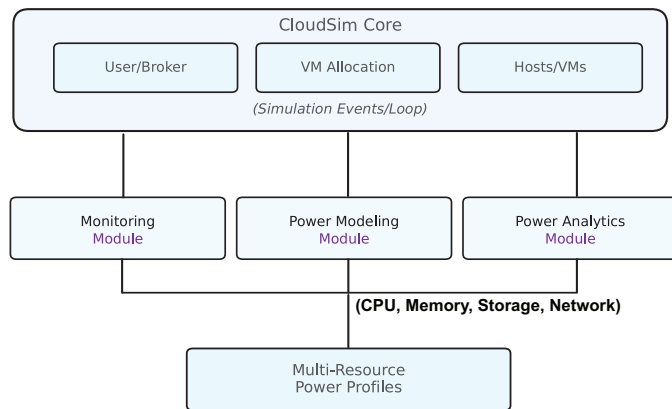


Figure 1. Architecture of the proposed multi-resource power modeling framework integrated into CloudSim.

The resulting subsystem-level power estimates are combined into a unified multi-resource power profile, which represents the dynamic host power behavior across all major subsystems. Interfaces are loosely coupled, allowing individual power models to be replaced, recalibrated, or extended without modifying CloudSim's internal scheduling or allocation logic.

B. Power Modeling Module

At each simulation timestep t , the total host power is computed as the aggregation of subsystem-level contributions:

$$P_{\text{host}}(t) = P_{\text{CPU}}(t) + P_{\text{RAM}}(t) + P_{\text{HDD}}(t) + P_{\text{NET}}(t) \quad (3)$$

Equation (3) reflects a structural decomposition of host power into major hardware subsystems. The formulation does not assume strict physical independence; rather, it provides a modular abstraction that avoids implicit bundling of non-CPU power into a single utilization variable. Subsystem models are parameterized and can be calibrated to represent different hardware platforms.

CPU Subsystem: In the proposed framework, the CPU model represents only the computational core's footprint. Non-CPU idle power is excluded to prevent double-counting across subsystems.

$$P_{\text{CPU}}^{\text{lin}}(t) = P_{\text{CPU, idle}} + (P_{\text{CPU, max}} - P_{\text{CPU, idle}})U_{\text{cpu}}(t) \quad (4)$$

$$P_{\text{CPU}}^{\text{spec}}(t) = \text{SPEC}_{\text{CPU}}(U_{\text{cpu}}(t)) \quad (5)$$

The CPU parameters are calibrated to represent CPU-only consumption rather than full-host measurements, ensuring consistency with 3.

Memory (RAM) Subsystem: Memory power is modeled as the sum of static refresh power and dynamic access-related power [16]:

$$P_{\text{RAM}}(t) = P_{\text{refresh}} \cdot M_{\text{alloc}} + P_{\text{active/GB}} \cdot M_{\text{active}}(t) \quad (6)$$

where M_{alloc} denotes allocated memory capacity and $M_{\text{active}}(t)$ denotes actively accessed memory at time t . The coefficients were selected based on values reported in prior measurement-based studies and scaled proportionally to the simulated hardware specifications.

Storage (HDD) Subsystem: Disk power is represented using a state-based abstraction:

$$P_{\text{HDD}}(t) = \alpha(t)P_{\text{active}} + \beta(t)P_{\text{idle}} + \gamma(t)P_{\text{startup}} \quad (7)$$

where $\alpha(t)$, $\beta(t)$, and $\gamma(t)$ denote the proportion of time spent in active, idle, and startup states during the sampling interval.

State proportions are estimated from simulated I/O throughput and request patterns. Alternative storage technologies (e.g., SSDs) can be incorporated by replacing the state parameters and power coefficients.

Network (NIC) Subsystem: NIC power is estimated using a throughput-efficiency relationship [6]:

$$P_{\text{NET}}(t) = P_{\text{nic, idle}} + \frac{R(t)}{\eta(R)} \quad (8)$$

where $R(t)$ represents network throughput and $\eta(R)$ denotes throughput-dependent energy efficiency. This formulation captures dynamic power scaling with traffic intensity while maintaining a configurable idle baseline. The efficiency function can be parameterized to reflect different link speeds and hardware generations.

C. Power Analytics Module

The Power Analytics Module aggregates subsystem-level power estimates to compute cumulative energy consumption at the VM, host, and datacenter levels. It supports detailed energy breakdowns by resource type, comparative

evaluation against baseline models, and workload-level energy profiling. This enables quantitative analysis of energy-aware scheduling, consolidation, and optimization techniques under heterogeneous workload conditions.

D. Integration with CloudSim

The framework is integrated into CloudSim as an extension layer without altering its core event handling, scheduling, or VM allocation mechanisms. Custom host and datacenter classes override native power accounting methods and redirect energy queries to the multi-resource aggregation function defined in (3), without modifying CloudSim’s core scheduling logic.

The Monitoring Module operates at fixed simulation intervals and retrieves utilization metrics directly from CloudSim entities, ensuring compatibility with existing VM placement and consolidation policies. This design preserves backward compatibility while enabling enhanced visibility into non-CPU energy consumption.

V. EXPERIMENTAL METHODOLOGY

This section details the simulation environment, datacenter configurations, and the stress-test workload designed to evaluate the behavioral differences between the proposed multi-resource framework and traditional CPU-centric baselines. The objective is comparative analysis of modeling structure rather than validation against a specific physical deployment.

A. Simulation Environment and Datacenter Setup

Experiments were conducted using CloudSim extended with our multi-resource modeling layer. To assess the framework across varying infrastructure scales, we simulated three heterogeneous datacenters (DC1–DC3), representing large, medium, and small environments, respectively. Each datacenter comprises an equal distribution of four server models, with hardware specifications detailed in Table I.

Subsystem-level power coefficients for CPU, RAM, storage, and network components were selected from measurement-based studies reported in the literature and scaled proportionally to the hardware capacities shown in Table I. The same coefficient configuration was consistently applied across all datacenters to ensure controlled comparison between modeling approaches.

TABLE I. HARDWARE SPECIFICATIONS OF THE FOUR SERVER MODELS USED IN ALL DATACENTERS (HP PROLIANT ML110 G3/G4/G5 AND A CUSTOM CELSIUS V840-BASED MODEL).

Model	Cores	MIPS	RAM (MB)	Storage (GB)	NIC (Mbps)
ML110 G3	2	6000	4000	160	1000
ML110 G4	2	3720	4000	160	1000
ML110 G5	2	5320	4000	146	1000
Celsius V840	4	10400	16000	80	1000

B. Workload Characteristics and Experimental Design

To evaluate the framework’s sensitivity to multi-dimensional resource demand, we utilize a high-intensity workload profile. Unlike standard CloudSim traces that primarily emphasize CPU-bound tasks, this experiment features a dynamic arrival pattern of 2,000 VMs configured with a 200% increase in baseline bandwidth and I/O requirements. This stress-test scenario is intended to force significant utilization of the RAM, Network, and Storage subsystems, thereby exposing structural limitations of models that ignore these components.

The workload is synthetically constructed to isolate subsystem-dominant behavior under controlled conditions. While real-world traces may exhibit more complex interdependencies, this design enables clearer observation of modeling bias under resource-intensive scenarios.

In this study, we maintain all hosts in an active state and explicitly disable power-saving optimizations (e.g., VM consolidation or DVFS). This experimental design allows us to quantify the baseline energy footprint of each modeling approach independent of scheduling effects, and to isolate differences arising purely from power estimation structure across the different datacenter scales summarized in Table II.

TABLE II. SUMMARY OF THE THREE SIMULATED DATACENTERS WITH EVENLY DISTRIBUTED SERVER TYPES.

Parameter	DC1 (Large)	DC2 (Medium)	DC3 (Small)
Total Servers	1200	600	400
Servers per Type	300 each	150 each	100 each
CPU Cores	3000	1500	500
RAM (GB)	8,400	4,200	2,800
Total BW (Gbps)	1,200	600	400

Simulations were executed over a 24-hour duration with a 5-second monitoring interval. This high-resolution sampling is critical for capturing transient spikes in memory and network activity, which are often smoothed over or lost in traditional 1-minute or 5-minute sampling windows.

C. Energy Computation

For the proposed MR model, host power is computed according to 3. Cumulative energy consumption (E) is derived from the instantaneous power samples using trapezoidal numerical integration to ensure numerical stability over the 24-hour simulation period:

$$E = \sum_{t=1}^{T-1} \frac{P(t) + P(t+1)}{2} \cdot \Delta t, \tag{9}$$

where $P(t)$ represents the power at time t and Δt is the 5-second sampling interval.

D. Evaluation Metrics

To quantify the comparative behavior of the modeling approaches, we report three primary metrics:

- **Total Datacenter Energy (kWh):** The aggregate energy consumption estimated by the Linear, SPECpower, and MR models.

- **Subsystem Energy Breakdown:** A granular percentage-based attribution of energy to CPU, RAM, Network, and Storage resources under the MR framework.
- **Relative Deviation:** The percentage deviation of CPU-only baselines relative to the MR reference model, calculated as:

$$\text{Deviation (\%)} = \frac{E_{\text{CPU-only}} - E_{\text{MR}}}{E_{\text{MR}}} \times 100. \quad (10)$$

The MR model is treated as a structurally richer reference for comparison rather than as empirical ground truth. Reported deviations therefore indicate differences in modeling abstraction rather than absolute physical measurement error.

VI. RESULTS AND DISCUSSION

This section evaluates the behavior of the proposed MR model against traditional Linear and SPECpower baselines under the high-intensity stress-test conditions previously defined. Rather than treating the MR formulation as physical ground truth, the analysis uses it as a structurally disaggregated reference model to examine how CPU-centric approaches behave when non-CPU activity is significant.

A. Datacenter Energy: CPU-only vs. Multi-Resource

Table III summarizes the total energy consumption for the three simulated datacenters (DC1–DC3), enabling a direct comparison of the relative deviation exhibited by single-resource models. Figure 2 visualizes these differences across infrastructure scales.

TABLE III. ENERGY CONSUMPTION AND RELATIVE DEVIATION OF CPU-ONLY MODELS COMPARED TO THE MR BASELINE.

DC	MR (kWh)	CPU-only Linear		CPU-only SPEC	
		Energy (kWh)	Error (%)	Energy (kWh)	Error (%)
DC1	881.4	903.4	+2.49	834.2	-5.35
DC2	721.4	739.5	+2.51	711.0	-1.44
DC3	716.5	734.4	+2.50	714.0	-0.35

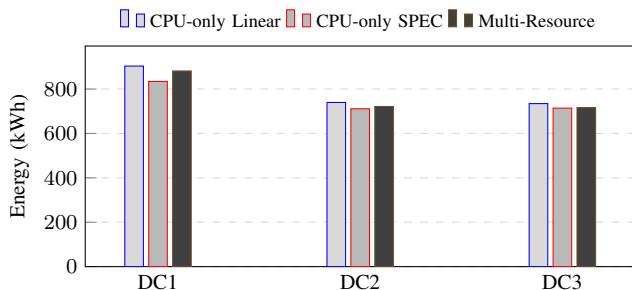


Figure 2. Comparative energy estimation across different models and datacenter scales.

Across all datacenter scales, the Linear baseline consistently shows a positive deviation of approximately +2.5%, while the SPECpower-only baseline yields a negative deviation reaching -5.35% in the most resource-intensive configuration (DC1). These results indicate a systematic directional bias in CPU-centric models under workloads with significant memory and I/O activity.

It is important to emphasize that these deviations are inherently workload-dependent. Under CPU-dominant scenarios with limited memory or I/O pressure, the gap between CPU-only and multi-resource formulations would be expected to narrow. The stress-test configuration used here intentionally amplifies cross-subsystem activity in order to expose structural modeling differences.

B. Subsystem Energy Attribution

The MR formulation enables granular subsystem-level attribution, as detailed in Table IV.

TABLE IV. SUBSYSTEM-LEVEL ENERGY BREAKDOWN (KWH / % CONTRIBUTION) UNDER THE MR MODEL.

Subsystem	DC1 (Large)	DC2 (Medium)	DC3 (Small)
CPU	592.1 / 67.18	483.4 / 67.01	480.3 / 67.03
Memory (RAM)	238.0 / 27.00	195.1 / 27.04	193.6 / 27.02
Storage (I/O)	43.5 / 4.94	36.6 / 5.08	36.3 / 5.07
Network (BW)	7.8 / 0.88	6.3 / 0.87	6.3 / 0.88
Total	881.4 / 100	721.4 / 100	716.5 / 100

While the CPU remains the dominant contributor (approximately 67%), memory and storage together account for roughly one-third of total energy consumption in this scenario. Because these subsystems do not scale strictly proportionally with CPU utilization, their independent dynamics explain the structural deviation observed in CPU-only host curves. The results therefore highlight modeling sensitivity to subsystem disaggregation rather than establishing absolute physical correctness.

C. Model Comparison and Directional Bias

Our evaluation reveals a distinct directional bias in traditional single-resource models. The Linear model serves as a pessimistic baseline, consistently overestimating total energy by approximately 2.5%. This stems from its simplified allocation logic, which treats idle and dynamic non-CPU power as a proportional extension of CPU load. Conversely, the SPECpower-only baseline provides a significant underestimation (reaching -5.35% in DC1). By modeling host power solely through CPU-centric benchmarks, it overlooks the dynamic fluctuations of RAM and network activity, which in our high-intensity scenario represent approximately one-third of the energy footprint.

D. Modeling Assumptions and Limitations

The proposed framework enhances modeling realism through multi-resource power functions and subsystem-level energy accounting, but it abstracts OS scheduling, hardware-level effects, and relies on integrated power models. Consequently, results should be interpreted comparatively rather than as precise real-world predictions. The enterprise workload traces used provide realistic correlated resource behavior, though they do not represent emerging paradigms such as GPU-accelerated, tightly coupled HPC, or serverless workloads. Resource utilization is measured only during active

execution, improving interpretability while excluding long-term idle effects, and the model accounts solely for IT power, omitting thermal and facility overheads. Furthermore, subsystem power coefficients are derived from literature-based measurements and scaled proportionally to simulated hardware configurations rather than calibrated against direct telemetry from a specific physical deployment. As such, the MR model serves as a structurally richer analytical reference rather than an empirically validated ground-truth baseline. Nonetheless, the controlled and deterministic experimental design ensures valid, reproducible, and meaningful relative comparisons across configurations.

E. Key Takeaways

- **Subsystem visibility:** Multi-resource modeling exposes non-CPU energy contributions (e.g., 289.3 kWh in DC1) that are structurally hidden in CPU-only formulations.
- **Directional modeling bias:** CPU-centric baselines exhibit consistent over- or under-estimation tendencies under I/O-intensive conditions.
- **Scale consistency:** Relative deviation remains largely scale-independent across DC1–DC3, indicating that the observed behavior is driven by modeling structure rather than datacenter size.
- **Workload sensitivity:** The magnitude of deviation is scenario-dependent and is expected to diminish under CPU-dominant workloads with limited memory and I/O activity.

VII. CONCLUSION AND FUTURE WORK

This paper presented a modular multi-resource power modeling framework for CloudSim that extends traditional CPU-centric energy simulation. By disaggregating host power into CPU, memory, storage, and networking components, the framework enables structured energy accounting and subsystem-level attribution within simulation studies.

Experimental results under high-intensity data-driven workloads indicate that traditional Linear and SPECpower-based models may deviate by up to approximately 5% when compared to the proposed multi-resource formulation. This deviation arises primarily from the exclusion of memory and I/O dynamics, which accounted for roughly one-third of total energy consumption in the evaluated scenario. These findings highlight the sensitivity of energy estimation to modeling structure rather than asserting absolute physical accuracy.

The framework is intended as a modular research instrument for subsystem-aware policy exploration. Future work may include hardware-level validation using empirical power measurements and telemetry traces to calibrate subsystem coefficients and strengthen realism, as well as support for heterogeneous accelerators and joint energy–performance optimization strategies that consider cross-subsystem interactions.

REFERENCES

- [1] I. E. A. (IEA), “Energy and AI – Analysis,” International Energy Agency, 2025, [Retrieved: 02, 2026]. [Online]. Available: <https://www.iea.org/reports/energy-and-ai/energy-demand-from-ai>
- [2] C. Jin, X. Bai, C. Yang, W. Mao, and X. Xu, “A review of power consumption models of servers in data centers,” *Applied Energy*, vol. 265, p. 114806, 2020, ISSN: 0306-2619. DOI: 10.1016/j.apenergy.2020.114806
- [3] L. Barroso and U. Hölzle, “The case for energy-proportional computing,” *Computer*, vol. 40, no. 12, pp. 33–37, 2007. DOI: 10.1109/MC.2007.443
- [4] A. Beloglazov, R. Buyya, Y. C. Lee, and A. Zomaya, “Chapter 3 - A taxonomy and survey of energy-efficient data centers and cloud computing systems,” in *Advances in Computers*, M. V. Zelkowitz, Ed., vol. 82, Elsevier, 2011, pp. 47–111. DOI: 10.1016/B978-0-12-385512-1.00003-7
- [5] R. Basmadjian, N. Ali, F. Niedermeier, H. De Meer, and G. Giuliani, “A methodology to predict the power consumption of servers in data centres,” in *Proceedings of the 2nd International Conference on Energy-Efficient Computing and Networking*, New York USA: ACM, 2011, pp. 1–10, ISBN: 978-1-4503-1313-1. DOI: 10.1145/2318716.2318718
- [6] J. A. Aroca, A. Chatzipapas, A. F. Anta, and V. Mancuso, “A measurement-based characterization of the energy consumption in data center servers,” *IEEE Journal on Selected Areas in Communications*, vol. 33, no. 12, pp. 2863–2877, 2015, ISSN: 1558-0008. DOI: 10.1109/JSAC.2015.2481198
- [7] 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, ISSN: 1553-877X. DOI: 10.1109/COMST.2015.2481183
- [8] R. Calheiros, R. Ranjan, A. Beloglazov, C. De Rose, and R. Buyya, “CloudSim: A toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms,” *Software - Practice and Experience*, vol. 41, no. 1, pp. 23–50, 2011. DOI: 10.1002/spe.995
- [9] R. Malhotra and P. Jain, “Study and comparison of cloudsims in the cloud computing,” *The SIJ Transactions on Computer Science Engineering & its Applications (CSEA)*, vol. 1, no. 4, pp. 111–115, 2013. DOI: 10.9756/SIJCSEA/V1I4/0104510201
- [10] B. Louis, K. Mitra, S. Saguna, and C. Åhlund, “CloudSimDisk: Energy-aware storage simulation in cloudsims,” in *2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC)*, 2015, pp. 11–15. DOI: 10.1109/UCC.2015.15
- [11] M. C. Da Silva Filho, “CloudSim Plus: A modern, extensible simulation framework for cloud computing and container-based infrastructures,” *IEEE Latin America Transactions*, vol. 15, no. 3, pp. 451–458, 2017. DOI: 10.1109/TLA.2017.7867605
- [12] M. H. Habaebi et al., “Extending CloudSim to simulate sensor networks,” *SIMULATION*, vol. 99, no. 1, pp. 3–22, 2023. DOI: 10.1177/00375497221105530
- [13] R. Muralidhar, R. Borovica-Gajic, and R. Buyya, “Energy efficient computing systems: Architectures, abstractions and modeling to techniques and standards,” *ACM Computing Surveys*, vol. 54, no. 11, pp. 1–37, 2022, ISSN: 0360-0300, 1557-7341. DOI: 10.1145/3511094
- [14] K. Farah, “Towards energy efficient cloud data centers - A framework for evaluation and analysis of energy efficiency,” [Retrieved: 02, 2026], Master thesis, Oslo Metropolitan University (OsloMet), 2024.
- [15] SPEC, *SPECpower_ssj 2008*, [Retrieved: 02, 2026]. [Online]. Available: https://www.spec.org/power_ssj2008/
- [16] M. Moreau, “Estimating the energy consumption of emerging random access memory technologies,” [Retrieved: 02, 2026], Master thesis, Institutt for elektronikk og telekommunikasjon, 2013. [Online]. Available: <https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2370830>