SysPower: Unified System-Wide Power Estimation for Energy-Aware Computing

Soren Lera[1], Benjamin James Carey[2], Wendell Conrad Richmond[3], Rinku Deuja[3], Nihat Altiparmak[3]
[1]*Hamilton College, USA* [2]*University of Kentucky, USA* [3]*University of Louisville, USA*
slera@hamilton.edu benjamin.carey@uky.edu {wendell.richmond,rinku.deuja,nihat.altiparmak}@louisville.edu

2026 IEEE International Conference on Cluster Computing (CLUSTER) | 979-8-3195-3093-6/26/$31.00 ©2026 IEEE | DOI: 10.1109/CLUSTER68337.2026.00059

***Abstract*—Accurate estimation of real-time power consumption is essential for energy-efficient computing systems and data centers. Existing Linux interfaces expose power information for only a subset of hardware components and therefore do not provide a unified view of total system power consumption. We present SysPower, a user-space framework that estimates system-wide power consumption by combining measurements from hardware telemetry sources with lightweight models for uninstrumented components. Unlike infrastructure-level monitoring solutions that rely on dedicated hardware and primarily target infrastructure management, SysPower provides a real-time estimate of system-wide power consumption together with subsystem-level power contributions using interfaces commonly available on existing systems. The resulting estimates are directly accessible to application and system software at a latency suitable for runtime energy-aware decision making. Evaluation against wall meter measurements shows a mean estimation error of 7.5% across diverse workloads while improving estimation accuracy relative to representative existing tools. SysPower is publicly available at `https://github.com/UOFL-CSL/syspower`.**

***Index Terms*—power monitoring, energy-aware computing.**

I. Introduction

Energy efficiency is a growing concern in modern computing systems and data centers, driven by the increasing scale of AI workloads and large-scale data center deployments. U.S. data centers currently consume 4-5% of national electricity generation and are projected to consume 9-17% by 2030 [1]. In Virginia, where data centers already account for approximately 25% of electricity consumption, this share is projected to increase to between 39% and 57% by 2030 [1]. These trends increasingly motivate energy-aware computing techniques that require timely and accurate power information.

Modern data centers often deploy hardware-based power meters such as metered PDUs that provide node-level power measurements. However, they are primarily designed for infrastructure monitoring rather than runtime decision making. They do not expose subsystem-level power contributions and are generally unavailable on desktops, laptops, workstations, and edge systems. Linux interfaces such as RAPL and device-specific telemetry provide only partial visibility into system power consumption. We address this gap with SysPower, a user-space framework that estimates system-wide power consumption in real time by combining hardware telemetry with lightweight models for uninstrumented components. A unified system-wide view is important because modern AI and data-intensive workloads exhibit strong energy interactions across CPUs, GPUs, memory, storage, and networking components [2]–[4]. Unlike existing approaches that focus on individual subsystems, SysPower provides a unified system-wide estimate while preserving per-component visibility and requiring no additional hardware. The proposed design enables immediate deployment across diverse systems while providing sufficient accuracy for energy-aware runtime decision making.

This research was supported in part by the U.S. National Science Foundation (NSF) under grant CNS-2349076.

II. Design and Implementation

SysPower estimates total system power by aggregating measured and modeled contributions from major hardware subsystems as follows:

$$P_{total} = P_{cpu} + P_{dram} + P_{gpu} + P_{io-devices} + P_{platform} \quad (1)$$

CPU power is obtained from Intel RAPL energy counters, while GPU power is obtained through vendor-provided telemetry interfaces such as the NVIDIA Management Library (NVML). DRAM power is obtained from available hardware telemetry when supported. Otherwise, DRAM, storage, and network power are estimated using the following bandwidth-normalized model:

$$P_{device} = P_{idle}^{device} + \left(P_{active}^{device} - P_{idle}^{device}\right) \cdot U_{device} \quad (2)$$

where power utilization of the device is calculated as $U_{device} = \min\left(\frac{BW_{cur}^{device}}{BW_{max}^{device}}, 1\right)$. For storage devices, BW_{cur}^{disk} is computed from block-layer statistics, while BW_{max}^{disk} is initialized from device-class performance characteristics and refined using observed peak bandwidth. For network interfaces, BW_{cur}^{nic} is computed from transmit and receive statistics, while BW_{max}^{nic} is derived from the negotiated link speed reported by the network interface. For DRAM, bandwidth is obtained from memory-controller statistics and normalized using device-class memory characteristics. Platform power accounts for motherboard, cooling, and power-delivery overhead. Motherboard power is modeled as a constant derived from platform specifications, fan power is estimated as $P_{fan,max}\left(\frac{RPM_{cur}}{RPM_{max}}\right)^3$ since fan power scales approximately cubically with rotational speed [5], and power-delivery losses are modeled as a constant derived from PSU efficiency and typical operating conditions.

The idle and active power characteristics used by the bandwidth-normalized device models are obtained from device telemetry when available and otherwise derived from device-class profiles. For example, some storage devices expose

2168-9253/26/$31.00 ©2026 IEEE
DOI 10.1109/CLUSTER68337.2026.00059

Authorized licensed use limited to: University of Louisville. Downloaded on October 23,2026 at 19:01:27 UTC from IEEE Xplore. Restrictions apply.

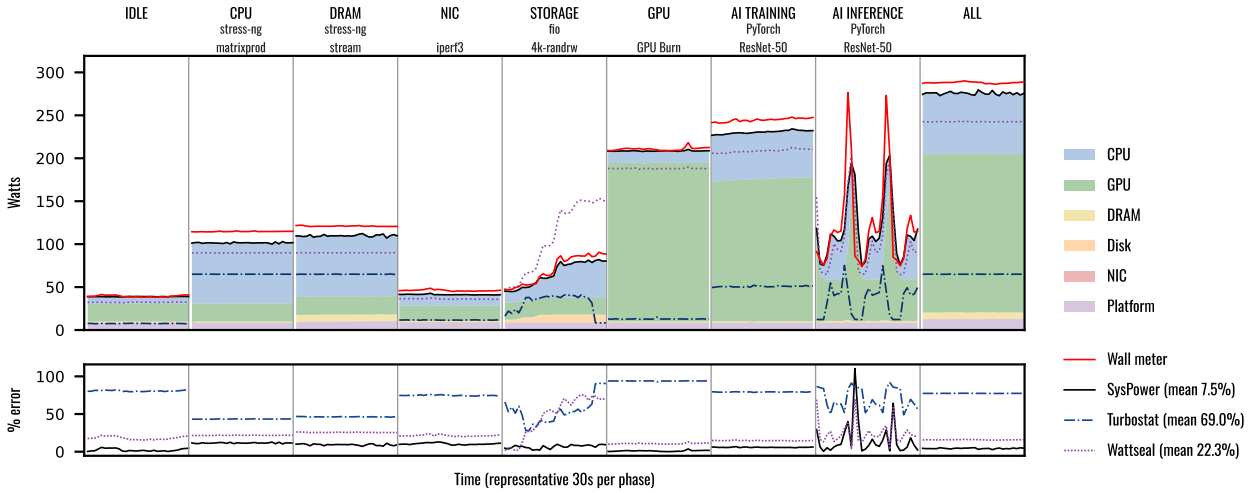

Fig. 1: Comparison of SysPower, Wall Meter (Ground-Truth), and Existing Tools

power-state descriptors through existing management interfaces, allowing SysPower to directly obtain or infer device power characteristics. In the absence of such information, SysPower falls back to built-in device-class profiles that define idle power, active power, and maximum bandwidth characteristics. Storage devices are classified according to their interface and media type (e.g., PCIe Gen3/Gen4/Gen5 NVMe SSDs, SATA SSDs, and SATA/SAS HDDs), while network devices are classified according to negotiated link speed. This approach leverages the observation that devices within the same class generally exhibit similar characteristics.

SysPower infers device classes using information already available in the Linux kernel, including block-layer attributes, storage driver subsystems, PCIe characteristics, and negotiated network link speeds. Device-class profiles provide portable default parameters that can be refined at runtime using observed peak bandwidth and administrator-provided overrides. The resulting per-device and subsystem estimates are aggregated into both a unified system-wide power estimate and a per-component power breakdown. Since the design relies exclusively on existing interfaces available on commodity Linux systems, it requires no kernel modifications, external instrumentation, or specialized hardware.

III. Evaluation

Experimental results were obtained using a system equipped with an Intel Core i7-12700 processor, an NVIDIA GeForce RTX 3060 GPU, 16 GB of RAM, a 1 TB SK Hynix PC801 NVMe SSD serving as the OS disk, a 2 TB Samsung 990 PRO NVMe SSD dedicated to storage experiments, and an Intel I219 1GbE NIC dedicated to network experiments. The system ran Ubuntu Server 26.04 LTS with Linux kernel version 7.0.0-27-generic. Ground-truth power measurements were obtained using a HOBO wall meter [6].

Figure 1 compares power estimates produced by SysPower against wall meter and two representative existing tools, Wattseal and Turbostat. We do not compare against tools such as Kepler, Scaphandre, or JoularJX because they primarily target application-level energy attribution or energy monitoring rather than unified system-wide power estimation, making direct comparisons inappropriate. The shaded regions in the figure show SysPower's estimated contributions of individual hardware subsystems to total system power. The evaluation includes idle measurements, workloads stressing individual hardware subsystems (CPU, DRAM, networking, storage, and GPU), AI training and inference workloads, and a composite workload that exercises all subsystems simultaneously. Results indicate that SysPower closely tracks ground-truth power consumption throughout workload execution, achieving a mean estimation error of 7.5% compared to 22.3% and 69% for Wattseal and Turbostat, respectively. Source code is available at `https://github.com/UOFL-CSL/syspower`.

IV. Future Work

Future work will explore integrating SysPower into the operating system kernel, enabling direct access to system-wide power estimates for energy-aware scheduling and resource management decisions. We also plan to evaluate SysPower across a broader range of systems and hardware platforms.

References

[1] Electric Power Research Institute, "Powering Intelligence 2026: Data Center Load Growth in Context," Electric Power Research Institute (EPRI), Tech. Rep., 2026. [Online]. Available: https://powering-intelligence.epri.com/load-growth.html

[2] B. Harris and N. Altiparmak, "Ultra-low latency ssds' impact on overall energy efficiency," in *12th USENIX Workshop on Hot Topics in Storage and File Systems (HotStorage '20)*. USENIX Association, July 2020.

[3] B. Harris and N. Altiparmak, "When poll is more energy efficient than interrupt," in *14th ACM Workshop on Hot Topics in Storage and File Systems (HotStorage '22)*, July 2022.

[4] H. Kim, G. Yeo, and M. Rhu, "Swarmio: Towards 100 million iops ssd emulation for next-generation gpu-centric storage systems," 2026. [Online]. Available: https://arxiv.org/abs/2604.06668

[5] Air Movement and Control Association, "AMCA Publication 201: Fans and Systems," 2023.

[6] *HOBO® Plug Load Logger (UX120-018) Manual*, Onset Computer Corporation, https://www.onsetcomp.com/sites/default/files/2023-04/17838-F%20MAN-UX120-018.pdf.

Authorized licensed use limited to: University of Louisville. Downloaded on October 23,2026 at 19:01:27 UTC from IEEE Xplore. Restrictions apply.