IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 2 March 2025, accepted 15 March 2025, date of publication 25 March 2025, date of current version 14 April 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3554728

RESEARCH ARTICLE

Single-Node Power Demand During AI Training: Measurements on an 8-GPU NVIDIA H100 System

IMRAN LATIF[1], (Member, IEEE), ALEX C. NEWKIRK[2], MATTHEW R. CARBONE[1], ARSLAN MUNIR[3], (Senior Member, IEEE), YUEWEI LIN[1], JONATHAN KOOMEY[4], (Member, IEEE), XI YU[1], (Member, IEEE), AND ZHIHUA DONG[1]

[1]Computing and Data Sciences Directorate, Brookhaven National Laboratory, Upton, NY 11973, USA
[2]Building Technology and Urban Systems, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA
[3]Department of Electrical Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL 33431, USA
[4]Koomey Analytics, San Francisco, CA 94011, USA

Corresponding authors: Imran Latif (ilatif@bnl.gov) and Alex C. Newkirk (acnewkirk@lbl.gov)

This work was supported in part by the Computational Resources of the Scientific Data and Computing Center, Brookhaven National Laboratory, under Contract DE-SC0012704, and in part by U.S. Department of Energy (DOE) Industrial Efficiency and Decarbonization Office (IEDO) under Contract DE-AC02-05CH11231.

ABSTRACT The expansion of artificial intelligence (AI) applications has driven substantial investment in computational infrastructure, especially by cloud computing providers. Quantifying the energy footprint of this infrastructure requires models parameterized by the power demand of AI hardware during training. In this work, we measured the instantaneous power draw of an 8-GPU NVIDIA H100 HGX node during the training of open-source image classifier (ResNet) and large-language models (Llama2-13b). We characterize power demand for a single node configuration, providing foundational data for future multi-node studies. The maximum observed power draw was approximately 8.4 kW, 18% lower than the manufacturer-rated 10.2 kW, even with GPUs near full utilization. Holding model architecture constant, increasing batch size from 512 to 4096 images for ResNet reduced total training energy consumption by a factor of 4. These findings can inform capacity planning for data center operators and energy use estimates by researchers. Future work will investigate the impact of cooling technology and carbon-aware scheduling on AI workload energy consumption.

INDEX TERMS AI training, sustainable computing, GPU power measurements, large language models.

I. INTRODUCTION

The modern era of deep learning requires both algorithmic innovation and dedicated hardware. As researchers identify scaling relationships in artificial intelligence (AI) applications between model scale, total training, and performance [1], computational hardware optimized for these workloads enables the training of enormous models with tens-to-hundreds of billions of parameters, using thousands of graphics processing units (GPUs). Beginning in late 2022, these steadily improving capabilities have captured consumer interest. Cloud infrastructure providers have responded with dramatic investment in hardware [2], both to meet this demand and in anticipation of further growth.

AI-optimized hardware, such as GPUs or tensor processing units (TPUs), are energy-efficient on a per-operation basis, and are enabling new capabilities in scientific and industrial settings [3], [4]. However, due to the scale of current and projected AI applications, this could still pose significant environmental costs through the operational energy footprint if not rigorously monitored [5]. While organizations training and deploying AI know their electricity consumption, any projections of future demand must model future load. Additionally, any actors without direct access to the relevant hardware, including electrical grid operators making capacity planning decision or researchers assessing sustainability, rely on these models. Energy-use models in turn rely on a

The associate editor coordinating the review of this manuscript and approving it for publication was Mario Donato Marino.

parameterized assumption of the power demand of hardware during AI training. To support these efforts to model energy use, we have measured the instantaneous power demand of current generation AI hardware across a variety of AI training workloads, and assessed the impact of model architecture and memory-batch size on energy performance.

A. CONTRIBUTIONS

The main contributions of this work are as follows:

- A power measurement methodology for AI training workloads on GPU-accelerated nodes;
- State-of-the-art experimental setup for accurate measurement of power for AI workloads;
- Rigorous experimental results of power measurement for modern AI workloads including results for a CNN-based deep learning model (ResNet) and a recent large language model, which may assist data center owners and operators in optimizing infrastructure provisioning for AI hardware.

B. SIGNIFICANT RESULTS

The most significant results of our work are summarized as follows:

- The maximum observed power draw during a maximum utilization stress test control was 8.4 kW, well below the 10.2 kW manufacturer rated maximum;
- During Llama training, median node power draw was 7.9 kW, with an average GPU utilization of 93%, *also* well below the rated maximum;
- Increasing ResNet training batch size from 512 to 4096 in image classifier training used 1 kW higher power on average, but only 25% of the total energy, due to reduced training time.

This work is structured as follows. In Section II, we begin with background on the growth of AI infrastructure and the need for accurate power measurements to inform energy use models. Section III details our methodology. Section IV documents the experimental setup including the hardware configuration, AI training workloads utilized, and the power measurement approach. Section V presents the power demand recorded across three training runs, varying in architecture and memory batching. Section VI discusses experimental results and the limitation of this work. Finally, Section VII concludes this work with the implications of these findings for data center capacity planning and energy estimation, and directions for future research.

II. BACKGROUND AND MOTIVATIONS

The commercialization of AI and the substantial hardware investment it requires have drawn increased attention to the energy footprint of these workloads [6]. Coinciding with heightened scrutiny from scholars and the public, companies have reduced the amount of data they release on hardware and metered energy [7]. This shift is partially due to the maturation of AI applications. As AI transitioned from an academic curiosity to a commercialized product, hardware operation details became a potential source of competitive advantage. Researchers internal to firms conducting training already have access to relevant aggregate data, e.g., their power bill. Several estimates of training energy intensity rely on this proprietary internal power draw data [8], [9], [10], [11]. Academic researchers instead have to rely on what data they have available, with one common approach to multiply the manufacturer rated power of AI-optimized GPUs or servers [12] by the reported training time (typically in GPU-hours) of a given model [13], [14], [15]. While this provides valuable bounds, it does not enable granular estimates of node-level power draw, assuming either full utilization of all non-GPU components or neglecting them entirely.

GPU component-level utilization is well-specified, in part because most AI-optimized hardware reports it through data center infrastructure management (DCIM) software. Alternative hardware for training AI models, such as application-specific integrated circuits (ASICs) or tensor processing units (TPUs) have similar performance benchmarking [16], [17]. However, node-level energy use is often approximated [18]. There is research grounded in metering of all components or entire nodes for previous hardware generations: Hodak et al. [19] found that GPUs to account for 70% of node power draw in training workloads; Wang et al. [20] found that wall-clock training time was the most consistent predictor of total energy consumption. Each of these studies were performed on previous GPU generations: the NVIDIA V100 and A100 respectively, with the latter finding GPU vintage to be a key determinant of total energy use.

Previous efforts on characterizing the distinct load profiles of physical modeling and AI workloads have been reported [21], [22], [23], though they typically focus on the chip/processor rather than node. Validating these estimates with node-level measured power serves to shed further light on the energy use of AI. While AI training and performance benchmarks have been proposed and implemented in the industry [24], they have not kept pace with commercial system scale-up [7]. Robust evidence should reflect recent hardware improvements and infrastructure investments. Additionally, there is an opportunity to compare traditional simulation workloads against AI training, as the two are performed on increasingly overlapping hardware configurations.

III. METHODOLOGY

Figure 1 depicts the overview of our methodology for benchmarking AI training power demands. We elaborate the key steps of our methodology in the following. Figure 1 shows our methodology's main steps in a generic manner and depicts its applicability to diverse AI workloads although in this work, we have run specific tasks/workloads to showcase our methodology. In our methodology, we run diverse AI workloads on AI hardware. We also run Burn tests in our methodology to stress the chips to the maximum. We then measure/benchmark power draw for running the AI workloads and burn tests. This process can repeat as needed as newer workloads are always introduced in AI and we need to consistently benchmark AI hardware for power consumption.

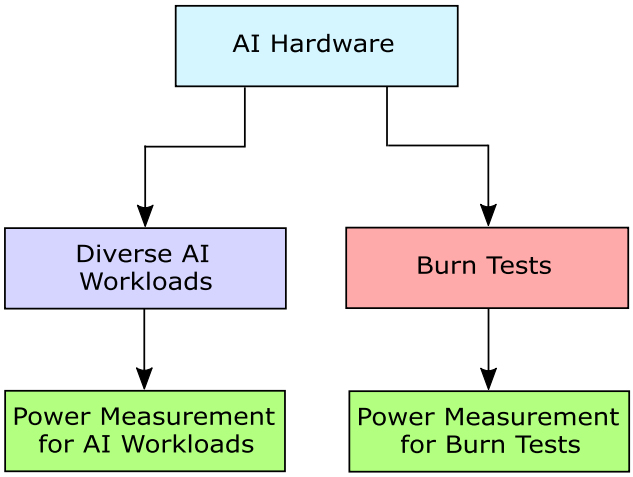


FIGURE 1. Overview of our methodology for benchmarking AI training power.

A. TRAINING WORKLOADS

We aim to characterize the power profiles of a diverse set of AI workloads. To ensure replicability and relevance to real-world applications, our methodology focuses on open-source models and datasets widely used in the machine learning (ML) community, such as by *MLPerf* [25]. While proprietary models and datasets used in production settings may differ in specifics, open-source workloads that are commonly used for benchmarking, testing, and research provide the best available approximation. As the computational load differs with architecture [21], we trained both convolutional neural networks (CNNs) for computer vision applications and transformer architecture large-language-models (LLM) for natural language processing. We selected Resnet and Llama2-13b as our workloads as they were *MLPerf Training v4.0* benchmarks.

B. AI HARDWARE

Modern ML workloads are typically executed on clusters of GPUs, assisted by CPUs for coordination and data transfer, as well as memory, storage, and interconnect infrastructure. We conducted our model training runs on an 8-GPU Nvidia H100 HGX node with 2 AMD EPYC 32-core CPU at the Brookhaven National Laboratory Scientific Data and Computing Center. At the time of experiment, the H100 was the most recently released NVIDIA GPU, and the configuration of the node was broadly consistent with that of the manufacturer recommendation.

C. BURN TESTS

To further contextualize AI training power results beyond our selected workloads, we performed hardware benchmarking stress tests known as a ''CPU burn'' and ''GPU burn'' which maximized utilization across all active computational components in the node: GPU, CPU, and memory. We conducted both isolated GPU stress testing and combined CPU-GPU-memory stress scenarios. This stress testing provides an upper bound on the power draw of the system under maximum load, serving as a control experiment for the ML workload measurements. This approach enabled us to contextualize the power consumption of AI workloads relative to the system's theoretical maximum demand.

D. BENCHMARKING POWER DRAW

Our methodology combines measurements from production-representative AI workloads with systematic hardware stress testing to provide a comprehensive view of system power consumption. The stress tests establish maximum power draw bounds, while the training workloads demonstrate typical operational demands on a training node. By comparing these measurements, we can evaluate how closely AI training scenarios approach theoretical power limits and identify potential optimization opportunities.

The following subsections detail our experimental setup, including the specific hardware configuration, software environment, model architectures, and datasets used, our methodology for data collection and analysis, and limitations of our approach.

IV. EXPERIMENTAL SETUP AND EQUIPMENT

A. COMPUTATIONAL HARDWARE

1) COOLING AND POWER INFRASTRUCTURE

Rear-Door Heat Exchangers (RDHx) cool the HGX node installed in a standard 19'' wide, 42U rack. Facility-chilled water at 16 °C flows through the RDHx to maintain a 24 °C ambient cooling temperature. Rack-mounted power distribution units (PDUs) deliver power to the HGX node through six standard C13 outlets. Overhead bus taps connect to the 208 V/3-phase, 50 A PDUs via twist lock mechanisms. A separate PDU in the same rack powers the RDHx unit.

2) CPU AND GPU INFORMATION

All experiments used AMD EPYC 9354 32-core processors, NVIDIA H100 GPUs with 80 GB of embedded on-chip memory, and were conducted on a single node with 8 GPUs, 2 CPUs, and 1.5 TB of memory.

B. TRAINING WORKLOAD AND SOFTWARE CONFIGURATION

We trained benchmark models to designed for two tasks: image classification and visual question answering (VQA).

1) IMAGE CLASSIFICATION

For the image classification experiments, we used the ResNet-152 [26] architecture trained on the CIFAR-10 [27] dataset. This dataset consists of 60k 32 × 32 color images spread across 10 categories (airplane, automobile, bird, cat, deer, dog, frog, horse, ship, and truck), with 50k images for training and 10k reserved for testing.

2) VQA

In the VQA experiments, we utilized LLaVA [28], a multi-modal large-language model, that combines a visual encoder, CLIP-L-14 [29], with a language-model decoder, Llama2-13b [30]. The visual encoder processes images,

while the Llama2 model, with ∼13 billion parameters, focuses on understanding and generating answers based on text. The VQA tasks were trained using a combined dataset of 665k pairs, including image-caption and image-question-answer pairs. These pairs were sourced from COCO [31], GQA [32], OCR-VQA [33], TextVQA [34], and Visual Genome [35], each providing rich visual-textual alignment. The COCO [31] dataset comprises approximately 118k training images, 5k validation images, and 40k test images, with resolutions ranging from 480×640 to 1024×768 pixels, and is primarily used for object detection, segmentation, and image captioning tasks. GQA [32] includes around 72k images paired with 943k questions, focusing on visual question answering with image sizes typically around 480×640 pixels. OCR-VQA [33] contains 50k images and questions, designed for text recognition within images. TextVQA [34] offers 28,408 images and 45,336 questions, requiring models to interpret text embedded in visual content. Visual Genome [35] provides 108k images, annotated with objects, attributes, and relationships, and is used for tasks like object detection, scene graph generation, and visual reasoning, with image sizes generally between 480×640 and 1024×768 pixels.

3) IMPLEMENTATION DETAILS

All tasks are implemented using Distributed Data Parallel (DDP) in PyTorch 2.2 [36]. For the image classification task, we train the ResNet model from scratch with a batch size of 512 and 4096 for 200 epochs, optimizing the entire model using stochastic gradient descent (SGD) with an initial learning rate of 0.1, a momentum of 0.9, and a weight decay of 5×10^{-4}. We employ a cosine annealing learning rate scheduler during training, which adjusts the learning rate based on a cosine function to enhance training performance and convergence. For the VQA task, we fine-tune the LLaVA [28] model on a combined dataset. The LLaVA [28] model comprises a visual encoder, a projector, and an LLM. Specifically, we use CLIP-L-14 [29] as the visual encoder, a two-layer multi-layer perceptron (MLP) with Gaussian error linear unit(GELU) activation as the projector, and Llama2-13b [30] as the LLM. During training, we keep the visual encoder and projector fixed and train only the LLM. The batch size is set to 128 with learning rate of 2×10^{-5} and warm up ratio is 0.03 with cosine learning rate scheduler.

4) GPU BURN

To evaluate the external validity and real-world applicability of our experimental results, we utilized the open-source GPU Burn tool [37]. A GPU burn is designed to stress test GPU setups by maximizing GPU memory utilization and compute load while validating calculation correctness. The tool forks a process for each GPU, allocates a high percentage of available memory, and performs continuous matrix multiplications using the CUBLAS library. We obtained the GPU-burn source code from the official GitHub repository at `github.com/wilicc/gpu-burn` and compiled it on our target Linux system following the provided instructions.

For our specific test scenario, we configured gpu-burn to utilize the default memory setting which is at 90% (72 GB) of available GPU memory (80 GB/GPU). Additionally, we enabled tensor core usage via the `-tc` flag to leverage the full computational capabilities of the GPUs and simulate peak compute demand. GPU Burn was executed in two configurations: first, as a standalone GPU stress test, and second, in parallel with the stress-ng CPU stress testing tool to assess performance under simultaneous CPU and GPU load. stress-ng, available at `github.com/ColinIanKing/stress-ng`, is a workload generation tool used for stress testing and benchmarking various components of a computer system, including CPUs, memory, disk I/O, and more. It provides a wide range of stress tests, allowing users to simulate different types of workloads to evaluate system performance and stability. The stress-ng code was ran with 256 tasks for 3D matrix operations. The run used 1.3 TB of 1.5 TB available host memory.

The combination of maximizing memory utilization, enabling tensor cores, and introducing CPU stress alongside the GPU burn aims to characterize the system's behavior under maximum-case computational demand.

C. POWER MEASUREMENT

The GPU power draw was measured using the Weights & Biases (wandb) [38] developer interface. The DCIM software tool, provided by NLYTE software, manages, monitors, and optimize the data center power and cooling infrastructure. Power and current data from the rack mounted PDU is pulled into the DCIM system over the network using communication protocols like SNMP and Modbus. The DCIM collects and processes this data, allowing operators to monitor, analyze, and optimize power usage at the rack level. The DCIM reports real-time data, including metrics like power draw, and alerts for potential issues related to critical infrastructure in the data center. This software dashboard also displays key performance metrics from the underlying compute system. One such metric is GPU utilization, which is harvested from the on-chip embedded sensor reported naively in the CUDA software environment.

DCIM software collects the available telemetry on the PDU and can alarm on specified metrics or on SNMP traps. Telemetry is collected from items such as, but not limited to power, current, voltage (line-to-line (L2L) and line-to-neutral (L2N)), information technology (IT) devices, servers, PDUs, transfer switches, uninterruptible power supplies, switch gear, transformers, branch circuit monitoring, and other end point devices.

We exported power demand and GPU utilization from the DCIM software our model training and hardware stress tests. These data included the minimum power demand, maximum power demand, and average power demand at each measurement interval. For the Llama training and the hardware stress tests, we collected data at 1-minute intervals. For the Resnet trainings, we collected data at 15 minute intervals. For GPU utilization, we exported the instantaneous power demand of each of the 8 GPUs in the node at each measurement interval. We averaged these values at each

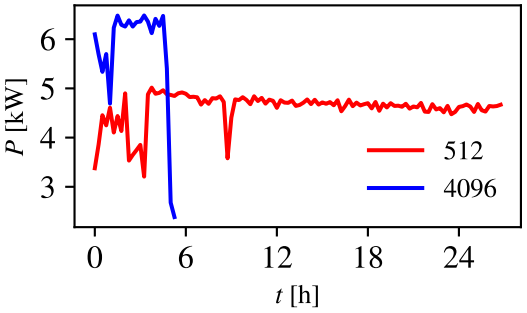


FIGURE 2. Instantaneous pre-training power demand measured in kW. Data corresponds to training ResNet with a batch size of 512 and 4096 images. Total energy usage for these two cases were computed by integrating the curves and are 123 kWh and 30 kWh, respectively.

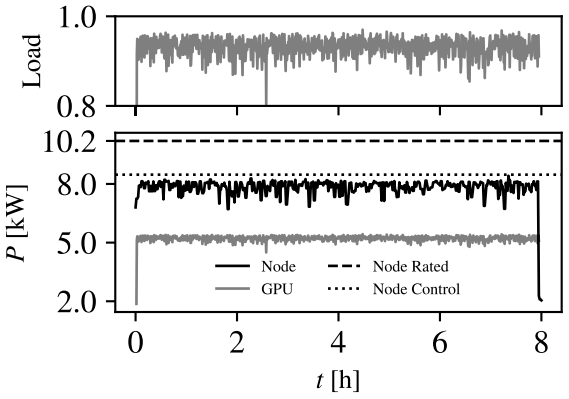


FIGURE 3. Instantaneous pre-training power demand measured in kW, and average total GPU load, for the Llama2-13b parameter model. The rated power of the system (10.2 kW) is shown as a horizontal dashed line. The median node power draw from the GPU+CPU burn control experiment is shown as a horizontal dotted line (at 8.43 kW).

timestamp and this average as a percentage of manufacturer-rated GPU power is our reported GPU utilization.

V. RESULTS

In this section, we report the observed ranges of power draws, examine the impact of factors like model architecture and batch size on energy usage, and analyze the relationship between GPU utilization and power demand. These results offer insights into the power requirements and energy footprint of AI training on contemporary hardware.

The training of the ResNet image classifier with a batch size of 512 images took over 26 hours. The median node power demand was 4.68 kW, with a maximum power demand measured during this training of 5.02 kW. This was followed by the training of ResNet with a batch size of 4096 images, which completed in 5 hours. The node demanded a median power draw of 6.26 kW during this workload, with a maximum measured demand of 6.48 kW. The instantaneous power demand for these experiments is showcased in Fig. 2.

The instantaneous power demand results for training Llama2-13b are shown in Fig. 3. Llama2-13b was trained with a batch size of 128 image question-answer pairs, completing in 8 hours. The median node power consumption was 7.92 kW, with a maximum measured power demand of 8.43 kW. GPU-level utilization during training (shown as the "GPU" label in Fig. 3) was close to maximum for all 8 chips, suggesting near computational saturation. It is noteworthy that our measured power draw never approached the manufacturer rated maximum of 10.2 kW ("Node Rated").

The dotted line in Fig. 3 ("Node Control") shows the total node power consumption during the maximally intensive GPU/CPU Burn stress tests, which maximized available GPU memory and tensor core operations. For this workload, where the GPU and CPU stress tests ran concurrently, the median power demand was 8.43 kW (with a max of 8.48 kW), only ∼400 W greater than the median demand of the Llama workload. The minimal difference in power consumption between the GPU stress test and the real-world Llama training workload demonstrates that our measurements offer a reliable characterization of the power requirements for computationally demanding workloads on this hardware.

VI. DISCUSSION

The power demand measurements in this study are specific to the particular combination of IT and cooling hardware in the tested system configuration, which may limit the generalizability of the findings. GPUs are consistent to within a product specification, but the supporting memory, supervisory CPUs, interconnect, and fan components vary between vendors. Accordingly, these results may be confounded by specific configuration of this node. However, all commercially available 8-H100 nodes have a manufacturer rated maximum power of 10.2 kW, meaning our finding of power draw at computational saturation should generalize. The core computational components (GPUs/CPUs) dominate power consumption when fully utilized, with supporting components like memory and interconnects contributing a relatively small fraction to total node power [21]. Additionally, while components generally operate within their specified bounds, the material variability of manufacturing logic devices means the precise power performance varies slightly chip-to-chip [39]. Future work should extend these results to include a greater variety of server configurations.

The total energy use and instantaneous power demand of a given training workload were sensitive to hyperparameter choice. The training task was held constant between ResNet-1 and ResNet-2, they only varied by batch size. The smaller batch size in ResNet-1 resulted in a 900 W decrease in node power demand, but increased total IT energy consumed by 90 kWh. When considering the total energy footprint of a given workload, IT power demand, runtime, and cooling efficiency all contribute.

Owing to the asymmetric costs for over- vs. under-provision of power for computational hardware (outage vs inefficiency), manufacturer-rated power limits typically serves as a design guideline for infrastructure procurement. The rack-level power distribution at the Scientific Data and

TABLE 1. Power was sampled at 1-minute intervals over the duration of training. Batch size is measured in images and tokens for the ResNet and Llama models, respectively. Parameters refers to the total model parameters for the trained model in millions or billions.

Training Workload	Model Task	Median Power (kW)	Batch Size	Duration (min)	Total Energy (kWh)	Average GPU Utilization (%)	Parameters
Idle Node	N/A	1.86	N/A	N/A	N/A	N/A	N/A
Resnet-512	Image	4.68	512	1,605	123	36	60M
Resnet-4096	Image	6.26	4,096	315	30	77	60M
Llama2-13b	VQA	7.92	128	480	62	93	13B
GPU Burn (no CPU)	Stress test	7.97	N/A	60	8	100	N/A
GPU Burn (+CPU)	Stress test	8.43	N/A	77	10.5	100	N/A

Computing Center supplies 70 kW of power to the racks which contain 5 Nvidia HGX nodes. These racks were designed according to the manufacturer rated maximum, with a margin of redundant power of approximately 27%. These results suggest that existing infrastructure could support an additional 8-GPU node while maintaining design-specified redundancy.

All training runs were conducted on a single node system, there is an opportunity for future work to extend these analyses to multi-node training. While multi-node training enables greater total computation, parallelization techniques are necessary to ensure and efficiency [40]. This multi-node parallelization would carry non-trivial communications cost, and any work should meter this power draw. Additionally, distributed training across multiple data centers should also be evaluated as this might expand opportunities for carbon-aware load shifting and scheduling, but could also potentially increase environmental footprint if data transmission or an increase in total training energy exceed associated carbon savings. Full visibility into the component-level share of node power draw is another opportunity for future research efforts. While technically challenging, detailed sub-metering of all active components in the node would identify the sensitivities of instantaneous power draw within different workloads, enabling hardware optimization.

These power data were collected using a single configuration of IT and cooling hardware. The active components within AI training servers include GPU's, supervisory CPU's, memory, storage drives, interconnect, and fans. While holding the active components of a given server constant, there may be interaction effects between the IT and cooling systems which could affect workload power demand. More effective cooling technologies, such as direct-to-chip liquid cooling, could allow the same computational utilization with lower fan usage. This would in practice be a shift of a greater share of the cooling services onto the more efficient system, enabling the same computational intensity at a lower system energy use. Future work should evaluate impact of cooling technology on node-level power demand, as well as other configuration, hyper-parametric, and infrastructure determinants of workload energy use.

Future work should expand this analysis across diverse hardware configurations to enable energy-optimized workload scheduling in heterogeneous cloud environments. While our measurements provide valuable baseline data for NVIDIA H100 systems, cloud providers typically operate mixed fleets spanning multiple GPU generations (e.g., A100, H100, B200) and node configurations within those generations. Characterizing power profiles across this hardware spectrum would enable training optimization based on both hardware availability and energy usage. Implementation would require careful accounting of interconnect energy consumption—both within clusters and across data center networks—as communication overhead can significantly impact total energy consumption in distributed training. Additionally, software-based frequency scaling and precision tuning could be investigated for their effect on power consumption and training time, potentially enabling finer-grained energy management without compromising model quality. These hardware-software co-optimization approaches could collectively yield substantial energy savings across large-scale AI deployments while maintaining computational performance.

VII. CONCLUSION

The increase in AI related computational demand has been enabled by investment in hardware infrastructure. This expansion has also prompted greater scrutiny into the energy footprint of this hardware. Estimates of AI energy use depend on parameterized value for in-workload power draw. To address this, we have measured the instantaneous power demand of a variety of AI and ML workloads on can 8 NVIDIA H100 HGX Node.

We found across all workloads, a maximum power demand of 8.48 kW, as compared to a manufacturer rated maximum power of 10.2 kW. This result was consistent with GPU utilization time-series data, which showed all 8 GPU's at or near maximum utilization during these time intervals. We also found that the total energy use of a given workload depends on memory batch size. We observed that a decrease in memory batch size for training an image classifier reduced average instantaneous power demand by 1 kW while increasing total pre-training energy use by a factor of 4. The findings from this research may assist data center owners and operators in optimizing infrastructure provisioning and enable them to deploy additional IT hardware using the same power distribution. Future work will investigate the impact of cooling technology on the energy use of a given workload, and evaluate the potential impact of carbon-aware scheduling on AI training workloads.

REFERENCES

[1] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling laws for neural language models," 2020, *arXiv:2001.08361*.

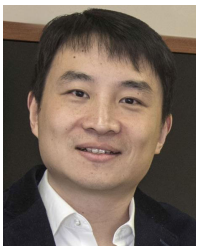
[2] N. Rattner and T. Dotan. (Sep. 2024). *The AI Spending Boom in Charts*. WSJ. [Online]. Available: https://www.wsj.com/tech/ai/artificial-intelligence-investing-charts-7b8e1a97
[3] B. Schmidt and A. Hildebrandt, "From GPUs to AI and quantum: Three waves of acceleration in bioinformatics," *Drug Discovery Today*, vol. 29, no. 6, Jun. 2024, Art. no. 103990.
[4] O. Nave and M. Elbaz, "Artificial immune system features added to breast cancer clinical data for machine learning (ML) applications," *Biosystems*, vol. 202, Apr. 2021, Art. no. 104341. [Online]. Available: https://api.semanticscholar.org/CorpusID
[5] L. H. Kaack, P. L. Donti, E. Strubell, G. Kamiya, F. Creutzig, and D. Rolnick, "Aligning artificial intelligence with climate change mitigation," *Nature Climate Change*, vol. 12, no. 6, pp. 518–527, Jun. 2022. [Online]. Available: https://www.nature.com/articles/s41558-022-01377-7
[6] E. Strubell, A. Ganesh, and A. McCallum, "Energy and policy considerations for deep learning in NLP," 2019, *arXiv:1906.02243*.
[7] J. Alben, "Computing in the era of generative AI," in *Proc. IEEE Int. Solid-State Circuits Conf. (ISSCC)*, vol. 67, Feb. 2024, pp. 26–28. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10454562
[8] P. Patel, E. Choukse, C. Zhang, Ì. Goiri, B. Warrier, N. Mahalingam, and R. Bianchini, "Characterizing power management opportunities for LLMs in the cloud," in *Proc. 29th ACM Int. Conf. Architectural Support Program. Languages Operating Syst.*, New York, NY, USA, Apr. 2024, pp. 207–222, doi: 10.1145/3620666.3651329.
[9] D. Patterson, J. Gonzalez, U. Hölzle, Q. Le, C. Liang, L.-M. Munguia, D. Rothchild, D. So, M. Texier, and J. Dean, "The carbon footprint of machine learning training will Plateau, then shrink," 2022, *arXiv:2204.05149*.
[10] C.-J. Wu, R. Raghavendra, U. Gupta, B. Acun, N. Ardalani, K. Maeng, G. Chang, F. Aga, J. Huang, and C. Bai, "Sustainable AI: Environmental implications, challenges and opportunities," in *Proc. Mach. Learn. Syst.*, vol. 4, Jan. 2022, pp. 795–813.
[11] D. Patterson, J. Gonzalez, Q. Le, C. Liang, L.-M. Munguia, D. Rothchild, D. So, M. Texier, and J. Dean, "Carbon emissions and large neural network training," 2021, *arXiv:2104.10350*.
[12] A. S. Luccioni, S. Viguier, and A. Ligozat, "Estimating the carbon footprint of Bloom, a 176B parameter language model," *J. Mach. Learn. Res.*, vol. 24, no. 253, pp. 1–15, Jan. 2022. [Online]. Available: http://jmlr.org/papers/v24/23-0069.html
[13] A. de Vries, "The growing energy footprint of artificial intelligence," *Joule*, vol. 7, no. 10, pp. 2191–2194, Oct. 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2542435123003653
[14] A. Lacoste, A. Luccioni, V. Schmidt, and T. Dandres, "Quantifying the carbon emissions of machine learning," 2019, *arXiv:1910.09700*.
[15] A. Faiz, S. Kaneda, R. Wang, R. Osi, P. Sharma, F. Chen, and L. Jiang, "LLMCarbon: Modeling the end-to-end carbon footprint of large language models," 2023, *arXiv:2309.14393*.
[16] N. P. Jouppi et al., "In-datacenter performance analysis of a tensor processing unit," in *Proc. ACM/IEEE 44th Annu. Int. Symp. Comput. Archit. (ISCA)*, New York, NY, USA, Jun. 2017, pp. 1–12, doi: 10.1145/3079856.3080246.
[17] N. P. Jouppi, C. Young, N. Patil, and D. Patterson, "A domain-specific architecture for deep neural networks," *Commun. ACM*, vol. 61, no. 9, pp. 50–59, Aug. 2018, doi: 10.1145/3154484.
[18] L. Bouza, A. Bugeau, and L. Lannelongue, "How to estimate carbon footprint when training deep learning models? A guide and review," *Environ. Res. Commun.*, vol. 5, no. 11, Nov. 2023, Art. no. 115014, doi: 10.1088/2515-7620/acf81b.
[19] M. Hodak, M. Gorkovenko, and A. Dholakia, "Towards power efficiency in deep learning on data center hardware," in *Proc. IEEE Int. Conf. Big Data*, Dec. 2019, pp. 1814–1820. [Online]. Available: https://ieeexplore.ieee.org/document/9005632
[20] X. Wang, C. Na, E. Strubell, S. Friedler, and S. Luccioni, "Energy and carbon considerations of fine-tuning BERT," 2023, *arXiv:2311.10267*.
[21] A. Govind, S. Bhalachandra, Z. Zhao, E. Rrapaj, B. Austin, and H. A. Nam, "Comparing power signatures of HPC workloads: Machine learning vs simulation," in *Proc. SC Workshops Int. Conf. High Perform. Comput., Netw., Storage, Anal.*, New York, NY, USA, Nov. 2023, pp. 1890–1893, doi: 10.1145/3624062.3624274.
[22] S. Shankar and A. Reuther, "Trends in energy estimates for computing in AI/machine learning accelerators, supercomputers, and compute-intensive applications," in *Proc. IEEE High Perform. Extreme Comput. Conf. (HPEC)*, Sep. 2022, pp. 1–8.
[23] Z. Zhao, E. Rrapaj, S. Bhalachandra, B. Austin, H. A. Nam, and N. Wright, "Power analysis of NERSC production workloads," in *Proc. SC Workshops Int. Conf. High Perform. Comput., Netw., Storage, Anal.*, New York, NY, USA, Nov. 2023, pp. 1279–1287, doi: 10.1145/3624062.3624200.
[24] S. Farrell et al., "MLPerfT HPC: A holistic benchmark suite for scientific machine learning on HPC systems," in *Proc. IEEE/ACM Workshop Mach. Learn. High Perform. Comput. Environ. (MLHPC)*, Nov. 2021, pp. 33–45. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9653178
[25] P. Mattson, V. J. Reddi, C. Cheng, C. Coleman, G. Diamos, D. Kanter, P. Micikevicius, D. Patterson, G. Schmuelling, H. Tang, G.-Y. Wei, and C.-J. Wu, "MLPerf: An industry standard benchmark suite for machine learning performance," *IEEE Micro*, vol. 40, no. 2, pp. 8–16, Mar. 2020.
[26] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[27] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep. TR-2009, 2009.
[28] H. Liu, C. Li, Y. Li, and Y. J. Lee, "Improved baselines with visual instruction tuning," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2024, pp. 26286–26296.
[29] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *Proc. Int. Conf. Mach. Learn.*, Jan. 2021, pp. 8748–8763.
[30] H. Touvron et al., "Llama 2: Open foundation and fine-tuned chat models," 2023, *arXiv:2307.09288*.
[31] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. Lawrence Zitnick, and P. Dollár, "Microsoft COCO: Common objects in context," 2014, *arXiv:1405.0312*.
[32] D. A. Hudson and C. D. Manning, "GQA: A new dataset for real-world visual reasoning and compositional question answering," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6693–6702.
[33] A. Mishra, S. Shekhar, A. K. Singh, and A. Chakraborty, "OCR-VQA: Visual question answering by reading text in images," in *Proc. Int. Conf. Document Anal. Recognit. (ICDAR)*, Sep. 2019, pp. 947–952.
[34] A. Singh, V. Natarajan, M. Shah, Y. Jiang, X. Chen, D. Batra, D. Parikh, and M. Rohrbach, "Towards VQA models that can read," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8309–8318.
[35] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *Int. J. Comput. Vis.*, vol. 123, no. 1, pp. 32–73, May 2017.
[36] A. Paszke et al., "PyTorch: An imperative style, high-performance deep learning library," 2019, *arXiv:1912.01703*.
[37] V. Timonen. (Oct. 2024). *Wilicc/gpu-burn*. [Online]. Available: https://github.com/wilicc/gpu-burn
[38] L. Biewald. (2020). *Experiment Tracking With Weights and Biases*. [Online]. Available: https://www.wandb.com/
[39] S. Hooker, "The hardware lottery," 2020, *arXiv:2009.06489*.
[40] J. Demmel, "Communication avoiding algorithms," in *Proc. SC Companion, High Perform. Comput., Netw. Storage Anal.*, Nov. 2012, pp. 1942–2000.

IMRAN LATIF (Member, IEEE) received the master's degree in mechanical engineering from The City College of New York. He is an Engineering Executive and the Chief Operations Officer with the Brookhaven National Laboratory, where he leads the Infrastructure Operations of High-Performance Computing Centers, supporting the global cutting-edge research collaborations on physics, life sciences, quantum, AI, and HPC. He has over 20 years of leadership experience delivering large scale engineering, construction, and sustainability projects. He leads worldwide sustainability research with leading academia and tech companies. His research work is focused on solving the real-life energy issues in the mission critical facilities and developing the AI and ML-based technologies to automate the sustainability processes. He is a subject matter Expert in direct to chip and immersion cooling in data centers.

ALEX C. NEWKIRK received the bachelor's degree in physics from Carleton College and the Ph.D. degree in engineering and public policy researching the criticality of semiconductors from Carnegie Mellon University. He is an Energy Technology Researcher with the Lawrence Berkeley National Laboratory and a member of the Center of Expertise for Energy Efficiency in Data Centers. He has led or contributed to research on organizational barriers to energy efficiency and AI related computational demand.

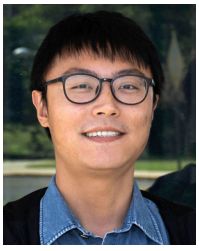
MATTHEW R. CARBONE received the B.S. degree in chemistry and the B.A. degree in physics from the University of Rochester and the Ph.D. degree in chemical physics from Columbia University. He is an Assistant Computational Scientist with the Computing and Data Sciences Directorate, Brookhaven National Laboratory. He was a United States Department of Energy Computational Science Graduate Fellow with Columbia University. He currently works in close collaboration with scientific user facilities on problems at the intersection of condensed matter physics/materials science and computational science, including machine learning modeling and autonomous experimentation.

ARSLAN MUNIR (Senior Member, IEEE) received the M.A.Sc. degree in electrical and computer engineering (ECE) from The University of British Columbia, Vancouver, Canada, in 2007, and the Ph.D. degree in ECE from the University of Florida, Gainesville, FL, USA, in 2012. He is currently an Associate Professor with the Department of Electrical Engineering and Computer Science, Florida Atlantic University. From 2007 to 2008, he was a Software Development Engineer with the Embedded Systems Division, Mentor Graphics Corporation (currently Siemens). He was a Postdoctoral Research Associate with the ECE Department, Rice University, Houston, TX, USA, from May 2012 to June 2014. His current research interests include embedded and cyber-physical systems, secure and trustworthy systems, parallel computing, artificial intelligence, and computer vision. He has received many academic awards, including the Doctoral Fellowship from the Natural Sciences and Engineering Research Council (NSERC) of Canada, the Gold Medal for Best Performance in Electrical Engineering, and the Gold Medals and Academic Roll of Honor for securing rank one in pre-engineering provincial examinations (out of approximately 300,000 candidates). His name is also included in the list of World's Top 2% Scientists compiled by experts at Stanford University based on the data from Elsevier's Scopus.

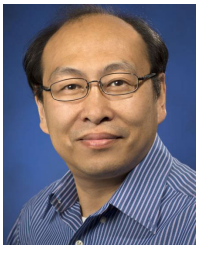


YUEWEI LIN received the B.S. degree from Sichuan University, Chengdu, China, the M.Eng. degree from Chongqing University, Chongqing, China, and the Ph.D. degree from the University of South Carolina, Columbia, SC, USA. He is currently a Senior Computational Scientist with the Brookhaven National Laboratory. He is also a Research Associate Professor with Stony Brook University. His research interests include machine learning, computer vision, and their applications within the scientific field.

JONATHAN KOOMEY (Member, IEEE) received the A.B. degree in history and science from Harvard University, and the M.S. and Ph.D. degrees from the Energy and Resources Group, UC Berkeley. He is the President of Koomey Analytics and is one of the leading international experts on the economics of climate solutions and the energy and environmental effects of information technology. He was a Staff Scientist with the Lawrence Berkeley National Laboratory, for many years. He has been a Visiting Professor/a Researcher with Stanford University, Yale University, and the University of California at Berkeley. He is the author or co-author of more than 200 articles, reports, and ten books, including *Turning Numbers into Knowledge: Mastering the Art of Problem Solving* and *Solving Climate Change: A Guide for Learners and Leaders*. More details can be found at koomey.com.



XI YU (Member, IEEE) received the master's and Ph.D. degrees in electrical and computer engineering from the University of Florida, in 2019 and 2022, respectively. He is currently a Postdoctoral Research Associate with the Computing and Data Sciences Directorate, Brookhaven National Laboratory. His research interests include domain generalization, robust machine learning, information theory, and scientific imaging applications.



ZHIHUA DONG received the Ph.D. degree in theoretical particle physics from Columbia University. He works on scientific code optimization, accelerated computing, and systems administration of HPC clusters. His current research focus is on performance portability of scientific applications.