Wafer-Scale Systems: A Carbon Perspective

ALICIA GOLDEN, Harvard University, USA
MARIAM ELGAMAL, Harvard University, USA
ABDULRAHMAN MAHMOUD, MBZUAI, UAE
GAGE HILLS, Harvard University, USA
CAROLE-JEAN WU, FAIR at Meta, USA
GU-YEON WEI, Harvard University, USA
DAVID BROOKS, Harvard University, USA

The rapid rise of Large Language Models (LLMs) has prompted a re-evaluation of system architecture design, making energy efficiency and sustainability more crucial than ever. Recently, wafer-scale architectures have emerged as a viable alternative for LLM training and inference, as evidenced by the success of Cerebras Systems. In this work, we examine the carbon implications of wafer-scale architectures as compared to traditional GPUs. As a case study, we examine LLMs on a Cerebras CS-3 system in order to quantify power and total carbon. Then, we analyze total carbon delay product (tCDP) to evaluate the carbon efficiency and performance potential of these systems. We take the first step towards exploring this trade-off for wafer-scale versus traditional GPU architectures – and ultimately find there exists a rich design space, depending on workload and hardware configuration.

CCS Concepts: • **Computer systems organization** → **Architectures**; • **Hardware** → **Power estimation and optimization**; **Impact on the environment**.

Additional Key Words and Phrases: Wafer-Scale, Carbon Footprint, Large Language Models, Sustainability, Sustainable Computing

1 Introduction

The recent surge in Large Language Models (LLMs) has forced datacenter architectures to adapt – prioritizing greater efficiency to handle increasing power and scale requirements. Datacenter energy consumption due to Artificial Intelligence (AI) alone is projected to increase to 6.7-12% of total US energy consumption by 2028 [32]. Meanwhile, the carbon emissions resulting from training LLMs is increasingly significant. These rising demands have catalyzed a shift toward rethinking both the infrastructure and compute hardware powering modern AI workloads.

The resulting push for extreme efficiency in the datacenter has introduced renewed interest in novel accelerators for AI. In recent years, a plethora of accelerators have come onto market, each introducing their own computational advantage, ranging from increased floating point operation (FLOP) count, specialized instructions, and novel memory systems [11, 22, 29]. One particular example is the *wafer-scale architecture*, where a chip consists of a single silicon wafer, thereby reducing the need for off-chip communication [28].

While wafer-scale architectures are not new [25, 28], they have only begun to show promise for large-scale applications in recent years. One example of such a wafer-scale system is the Cerebras architecture, first debuted in 2019 as the largest processor ever built.

Authors' Contact Information: Alicia Golden, Harvard University, Cambridge, MA, USA; Mariam Elgamal, Harvard University, Cambridge, MA, USA; Abdulrahman Mahmoud, MBZUAI, Abu Dhabi, , UAE; Gage Hills, Harvard University, Cambridge, MA, USA; Carole-Jean Wu, FAIR at Meta, Cambridge, MA, USA; Gu-Yeon Wei, Harvard University, Cambridge, MA, USA; David Brooks, Harvard University, Cambridge, MA, USA.

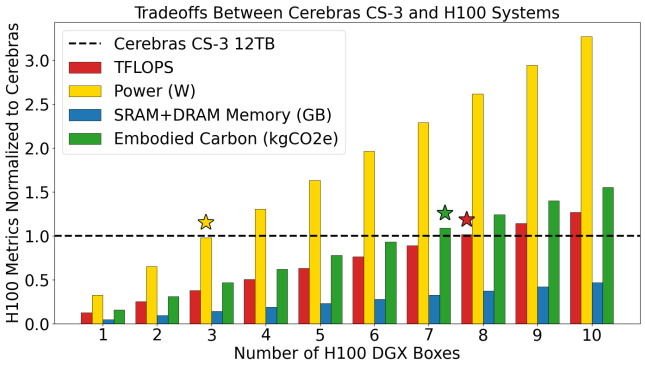


Fig. 1. Wafer-Scale systems present distinct performance, power, memory, and embodied carbon trade-offs as compared to traditional GPUs. Here we compare a case-study of CS-3 versus H100 systems, sweeping the number of H100 DGX boxes *(x-axis)* and plotting each metric normalized to Cerebras CS-3 system *(y-axis)*. For example, taking the ISO-FLOPS scenario represented by the red star (8 H100 DGX boxes versus a Cerebras CS-3), the H100 choice has 2.61× more power, 2.69× less GB SRAM and DRAM, and incurs 1.24× the embodied carbon.

Their subsequent family of wafer-scale engines (WSE) are manufactured to be 46,000 mm^2 processor chips, each containing up to 2.6 trillion transistors [21]. The most recent Cerebras CS-3 system in particular offers a promising alternative to traditional GPU architectures – achieving over 19× faster Llama4-Scout inference speeds as compared to H100 GPUs [4].

Figure 1 examines an example of a wafer-scale system (the Cerebras CS-3 chip) and compares the trade-offs in terms of performance, power, memory, and embodied carbon to a traditional GPU architecture (i.e., H100 GPU). We sweep the number of H100 DGX boxes and compare the FLOPS, power, and memory available (SRAM+DRAM) in the system [2, 3, 7]. We then compute the embodied carbon, as is later described in Section 3. Each star represents when the H100 system metric is within 4% of the Cerebras value. We find that for the same amount of power, an H100 system of 3 DGX boxes would offer 0.37× FLOPS, 0.13× memory capacity, and 0.46× embodied carbon compared to CS-3. However, for approximately the same number of FLOPS, H100 systems take 2.61× more power than Cerebras, while using 2.69× less memory and consuming 1.24× embodied carbon.

However, this is only the tip of the iceberg. In order to fully analyze the design trade-offs for wafer-scale architectures as compared to traditional GPUs, we must (i) understand how existing commercial wafer-scale systems, such as Cerebras, fit into the wafer-scale design landscape, (ii) characterize performance of LLM workloads

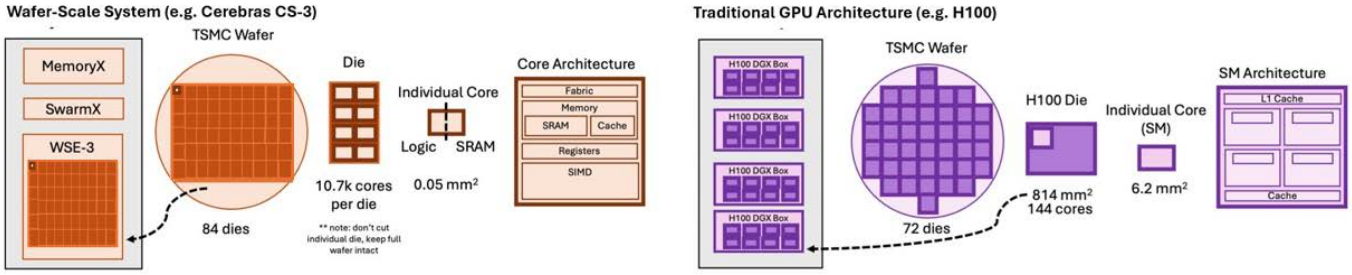


Fig. 2. Architecture diagrams of wafer-scale system (e.g. Cerebras CS-3) as compared to traditional GPU system (e.g. H100 DGX system). Note that for wafer-scale systems, individual dies in the wafer are physically connected, whereas in traditional GPU manufacturing, individual dies are diced and then re-packaged together [1, 10, 23]. Diagrams are not drawn to scale.

on these systems, and (iii) analyze the full carbon-performance design space via careful evaluation of carbon efficiency metrics, such as tCDP [13, 14]. We address each in turn throughout the paper.

While this is the first work to the best of our knowledge to analyze carbon of wafer-scale, previous works have analyzed the carbon footprint of traditional and emerging computing systems [15–17, 27, 30, 35, 37]. Furthermore, prior works have explored a variety of energy efficiency and carbon footprint optimizations across the stack. For instance, DynamoLLM optimizes energy and operational carbon emissions of LLM inference environments [34]. Junkyard Computing optimizes the computational carbon intensity (CCI), accounting for both embodied and operational carbon, to repurpose old smartphones for cloud-scale computations [36]. Other works, such as Carbon Explorer and EcoServe, implement carbon-aware scheduling and AI inference resource provisioning in datacenters [8, 19]. Additionally, from a hardware design perspective, CORDOBA enables designers and architects to optimize computing systems for carbon efficiency, quantified using total carbon-delay product (tCDP), despite uncertainties in carbon modeling data [14].

In this paper, we build upon the aforementioned techniques to present an analysis comparing carbon footprint of wafer-scale to traditional GPU architectures. We use Cerebras as a case study here, however, our analysis can be extended to other wafer-scale systems. The key contributions of this work are:

(1) **We quantify carbon footprint trade-offs of wafer-scale architectures versus traditional GPU architectures.** Depending on the memory configuration used, we find that the embodied carbon of Cerebras systems is between 0.34-1.86× the embodied carbon of the iso-flops equivalent H100 DGX box system (Section 3).

(2) **We characterize the power consumption of Cerebras CS-3 systems across a range of LLM models**, and find that typical power consumption during model training is 21% higher than idle power (Section 4).

(3) **We evaluate the carbon-aware design space in terms of total carbon footprint and tCDP and find that the optimal configuration depends not only on hardware type but also on how the system is deployed over time – i.e., the amount of work done and time the system is active.** While not initially carbon-efficient, wafer-scale architectures like Cerebras CS-3 can become more carbon-efficient than GPU-based systems when used actively for a large enough fraction of their lifetime. For example, if a Cerebras CS-3 system is active >40% of the time throughout a three year lifetime, and given the assumptions stated in Section 5.2 and Section 6, it can be at least 1.54× more carbon-efficient (quantified by tCDP) than the corresponding iso-flops system of 8 H100 DGX boxes when processing the same amount of work (Section 5).

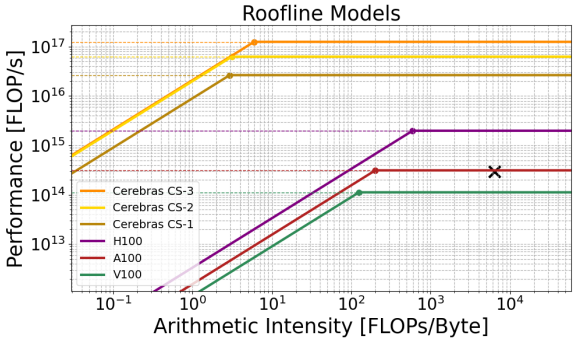


Fig. 3. Comparing wafer-scale and GPU Architectures. Roofline shows Cerebras enables traditional memory-bound workloads with arithmetic intensity ranging from 10-100 FLOPs/Byte to be compute-bound. X marks a Llama workload, calculated analytically.

2 Wafer-Scale Architectures

In this section, we detail the architectural differences of wafer-scale systems compared to traditional GPUs.

2.1 Understanding Wafer-Scale

Wafer-scale chips consist of a single silicon wafer, eliminating the need for small individual chips to be manufactured separately and then integrated together on package. This offers advantages in terms of fast and more energy-efficient die-to-die connections [21, 28]. Additionally, the high memory bandwidth for on-chip memory enabled by wafer-scale can help to alleviate the traditional memory wall for memory-bound workloads [28].

2.1.1 Cerebras Chip Design. One promising example of the viability of wafer-scale architectures is Cerebras Systems. Cerebras' most recent generation features the WSE-3 chip, which is 57× larger in area as compared to a single H100 GPU [3]. The wafer contains more than 900,000 identical cores, each consisting of logic to SRAM in a 50:50 ratio [20, 23]. Altogether, the WSE-3 has 44 GB of SRAM on chip, which reduces the need for off-chip communications traditionally performed with NVIDIA's NVLink and Infiniband networks [3].

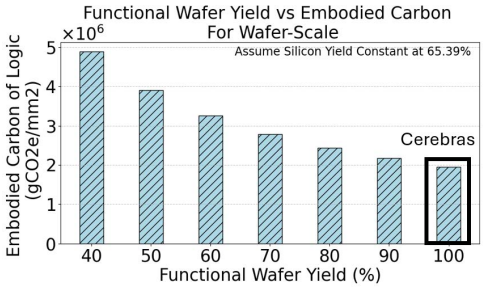


Fig. 4. Sweep of functional wafer yield *(x-axis)* to capture range of resulting embodied carbon *(y-axis)*. We assume 65.39% silicon yield of a 5 nm wafer.

2.1.2 Memory and Storage. In addition to the WSE-3 chip, Cerebras systems include Swarm-X and Memory-X, specialized network and memory components for effective communication and storage of large model parameters, respectively. Swarm-X is an active network fabric that performs broadcast and all-reduce operations while connecting across WSE-3 chips. In particular, Swarm-X allows Cerebras to achieve near-linear performance scaling across multiple wafers [33]. Memory-X consists of a combination of Flash and DRAM, which holds model weights. In a typical training run, model weights are stored in Memory-X before being streamed into the wafer where the activations reside. The gradients are subsequently streamed out of the wafer back to Memory-X where optimizer update happens [33].

As shown in Figure 2, wafer-scale architectures consist of the largest rectangle of silicon area that fits inside the circular wafer. While individual die exist, they are not diced as in traditional semiconductor chips. Instead, the wafer stays as one chip. We further examine the yield implications of this approach in Section 3.

2.2 Comparison to GPU Architecture

Figure 2 additionally highlights a typical GPU architecture, using H100 as an example. Starting with the same circular wafer, individual die on the wafer are then diced. Each die represents an H100 chip, which is assembled with other chips for a DGX box. A DGX box consists of 8 H100s connected by high-bandwidth NVLink.

In order to examine how wafer-scale architectures such as Cerebras compare to traditional GPUs, we first note that both H100 and CS-3 are manufactured with TSMC 5nm technology node, whereas A100 and CS-2 are both 7nm. Then, we examine peak performance and memory bandwidth. As the roofline in Figure 3 shows, Cerebras CS-3 system features 63× higher peak FLOPS than an H100 GPU. The large amount of on-chip SRAM with high memory bandwidth allows the memory-bound portion of the curve to be shifted left, thereby enabling workloads that are traditionally memory bound on GPUs to be compute bound on Cerebras. We note that both memory- and compute-bound workloads have potential to benefit from wafer-scale architectures, although the exact performance differences would depend on the properties of the workload.

Table 1. Embodied Carbon Calculations

Metric	H100	Cerebras CS-3
Total Wafer Area (mm^2)	70685.83	70685.83
Total Area Used (mm^2)	58,608.00	46,225.00
Silicon Yield (%)	82.91%	65.39%
$C_{embodied}$ Logic (gCO_2e/mm^2) [18]	29.15	
$C_{embodied}$ DDR4/LPDDR5 (gCO_2e/GB) [19]	290.00	
$C_{embodied}$ GDDR6 (gCO_2e/GB) [19]	360.00	

3 Quantifying Embodied Carbon of Wafer-Scale

In this section, we detail our methodology to analyze carbon footprint of wafer-scale systems as compared to traditional GPU architectures. We present a generalized analysis of wafer-scale systems before quantifying carbon for our case study of Cerebras.

Yield. First, we need to understand the yield of wafer-scale architectures and its implications on quantifying embodied carbon. We note that historically, wafer-scale systems were impractical due to yield implications [25, 28]. We quantify yield of wafer-scale architectures in two parts. First, we calculate silicon yield defined by the total silicon used per wafer divided by total wafer area. Second, we incorporate the "functional wafer yield" (which we define here as the fraction of wafers that result in a functional chip). We follow the formula below to calculate total yield:

$$\text{Yield} = (\text{Silicon Yield}) \times (\text{Functional Wafer Yield})$$

There are a plethora of models for determining yield in the literature [12, 26]. In order to minimize assumptions, we sweep the percentage of functional wafer yield and quantify the resulting embodied carbon. This provides a range of embodied carbon values for wafer-scale architecture, without beholding to a specific manufacturing or assembly technique. Figure 4 illustrates this sweep. We show the embodied carbon of wafer-scale architectures on a 5 nm wafer when sweeping functional wafer yield from 40-100%.

To calculate embodied carbon ($C_{embodied}$), we assume a 300 mm TSMC 5 nm wafer and the Taiwan energy grid [17] while taking the above yield into account. We use IMEC reported gCO_2e/mm^2, assuming no abatement [18]. The parameters for our calculations are in Table 1.

Techniques to Improve Yield. Recent innovations have improved die yield significantly. For example, Cerebras enables wafer scale integration by utilizing small-sized cores and high defect tolerance as well as a technique to adaptively route around bad cores [10]. Since the area of each individual WSE-3 core is small (0.05 mm^2), the resulting silicon area lost due to each defect is considerably less as compared to a traditional GPU, where an entire Streaming Multiprocessor (SM) unit (6.2 mm^2) would be lost. As outlined in [10], Cerebras assumes random die placement such that each defect lands in a unique core, and then calculates the resulting die space lost due to defects. Cerebras' routing architecture allows the system to route around defective cores to maintain the chip's computational capabilities and high fault tolerance. Current CS-3 products have 900,000 active cores out of 970,000 physical cores, i.e. 93% of the chip silicon area is active. Under these assumptions, the functional wafer yield of CS-3 becomes close to 100%, thereby falling on the lowest end of the embodied carbon sweep from Figure 4.

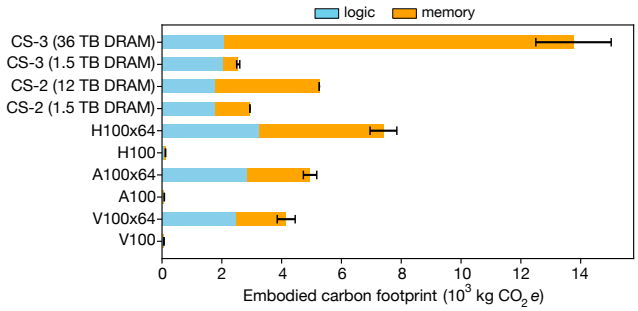


Fig. 5. Embodied carbon footprint for a range of Cerebras and NVIDIA systems, accounting for logic die and memory. We find the embodied carbon of the minimum CS-3 memory configuration is 22× larger than a single H100 GPU, but 2.9× smaller than an iso-FLOPS system of 8 DGX boxes.

Embodied Carbon of Wafer-Scale Logic and Memory. We expand our above analysis quantifying embodied carbon of logic to now include embodied carbon of memory used in a wafer-scale system and GPU architectures. We use CS-2 and CS-3 architectures as an example, and quantify the resulting embodied carbon from the logic and the Memory-X system, specifically for DRAM to ensure fair comparison with GPU architectures. There are several Memory-X DRAM capacities available, and so we sweep the minimum (1.5 TB) and maximum (36 TB) capacities. In a similar manner, we look at three generations of NVIDIA GPU architectures and quantify embodied carbon of the logic and HBM. We repeat this analysis for a system of 8 DGX boxes (64 GPUs) to analyze the iso-FLOPS scenario from Figure 1.

Figure 5 shows the resulting embodied carbon footprint of both logic and memory for the range of Cerebras systems and NVIDIA GPU systems. We find the embodied carbon of the minimum CS-3 memory configuration is 22× larger than a single H100 GPU, and 2.9× smaller than a iso-FLOPS system of 8 DGX boxes.

4 Examining the Cerebras CS-3 System

To better understand the power and energy consumption of wafer-scale architectures, we profile a Cerebras CS-3 system across LLM workloads.

Power Traces Methodology. We record power of a CS-3 system through measurement of its power distribution units (PDUs). Each CS-3 system contains 9 PDUs, which we sum to get total power. Our resulting power measurements represent power of the WSE chip, not the Memory-X and Swarm-X systems. We work in collaboration with Pittsburgh Supercomputing Center (PSC) and CerebrasCloud to access Cerebras hardware and obtain power measurements [9]. Through a series of experiments, we observe idle power is 19.7 kW.

Power Measurements Results. Figure 6 shows a snapshot of Gemma2-27B model training and the resulting power trace. We highlight three sections of our training snapshot here, including power for device setup, power during 2 training steps, and power during device checkpointing. We find that power reaches up to 24.1 kW during model training, as highlighted in green. We observe that during device setup, power increases slightly from idle power due to transfer of weights, and power consumption remains near idle power when saving the model checkpoint.

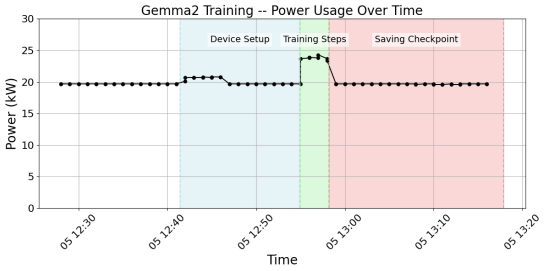


Fig. 6. Gemma2-27B training power trace on Cerebras CS-3. We analyze the device setup, training steps, and saving checkpoint phases separately, as shown. Power during model training reaches a peak of 24.1 kW.

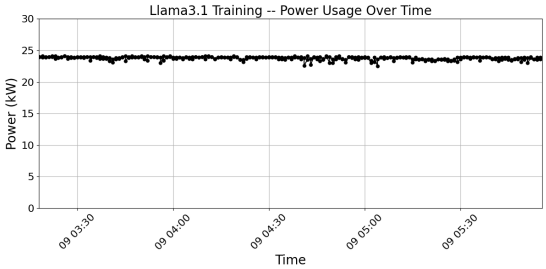


Fig. 7. Llama3.1-8B training power trace on CS-3 system. Power draw for training is relatively constant at ~24 kW.

We put this snapshot into context by profiling longer training runs, such as the Llama3.1-8B power trace in Figure 7. We find that power consumption during model training stays relatively constant at about 24 kW. From these experiments, we see that typical power consumption during model training is 21% higher than idle power. In contrast, typical power consumption for training on H100 GPU is 9.29× higher than idle power. This has potential implications for operational carbon, as we will examine in the next section.

5 Carbon-Aware Design Space

We subsequently explore the carbon design space to quantify the trade-offs between wafer-scale systems and traditional GPU architectures. To do so, we analyze both (i) total carbon (embodied + operational) and (ii) total carbon delay product (tCDP) [14].

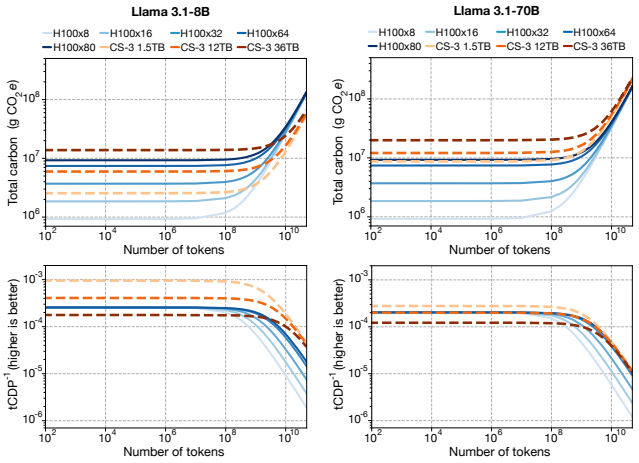
5.1 Operational and Embodied Carbon

We analyze total carbon emissions of wafer-scale versus GPU architectures by quantifying their embodied and operational carbon. We continue to use the CS-3 system as an example of a wafer-scale chip and our methodology can be applied to other wafer-scale systems.

Embodied carbon calculations are derived as shown in Section 3. Operational carbon is calculated using an upper bound of power for each system (e.g., measured training power for Cerebras and TDP for H100, as further discussed in Section 6), along with the duration of execution of an LLM workload and a carbon intensity of 380 kgCO_2e/kWh [17].

5.2 Quantifying Carbon Efficiency

To quantify carbon efficiency and capture the carbon-performance trade-off of wafer-scale systems, we examine tCDP, which is the



(i) Llama 3.1-8B (ii) Llama 3.1-70B

Fig. 8. Total carbon footprint *(top)* and carbon efficiency *(bottom)* of different H100 GPU configurations and CS-3 systems for projected (i) Llama 3.1-8B and (ii) Llama 3.1-70B.

product of total carbon and workload execution time [14]. Here, we quantify execution time $= \frac{\text{number of tokens generated}}{\text{tokens per second}}$.

We compare total carbon and tCDP for CS-3 versus traditional GPU architectures for the task of LLM inference. In order to create a fair comparison, we sweep number of H100 DGX boxes, as was done in Figure 1. We use reported benchmark numbers from [5, 6] to capture Cerebras performance in tokens/second as a throughput metric (note: not a latency target). We assume each platform runs with the optimal batch size to maximize their respective memory capacities. We additionally assume the number of WSE-3 wafers necessary based on memory capacity as outlined in [23]. We report three Memory-X DRAM configurations of CS-3 for completeness. We then compare CS-3 performance to reported performance of H100 systems assuming 1 DGX box [5, 6].

To scale to multiple DGX boxes, we use a FLOPS calculation to project tokens/sec based on combined peak FLOPS of each system. Note that we use this as a theoretical performance projection to project similar tokens/sec onto a larger amount of hardware (e.g., as an upper bound for inference serving), since we acknowledge this is independent of Llama-8B's ability to parallelize over the hardware specifically. We compare tCDP, a measure of carbon efficiency, across two common LLM sizes (Llama3.1-8B and Llama3.1-70B) to quantify the impact of increasing number of WSE-3 wafers needed.

5.2.1 Running Continuously. We first examine the resulting total carbon and tCDP assuming we run the system continuously to generate a given number of tokens. We sweep the number of tokens generated up to $O(10^{11})$, which represents running continuous inference for approximately 1.5 years. Figure 8i highlights the resulting total carbon and tCDP analysis for the 8B parameter model (i.e., using 1 WSE-3 wafer). We find that the CS-3 1.5 TB system initially has higher total carbon than 1 or 2 H100 DGX boxes, until generating $>10^9$ tokens at which point total carbon of CS-3 is less. We note that CS-3 systems have higher $tCDP^{-1}$ (higher values correspond to better carbon efficiency) regardless of the number of tokens generated

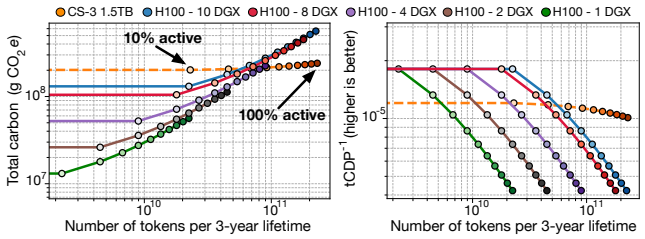


Fig. 9. Total carbon *(left)* and tCDP *(right)* plotted versus number of tokens generated *(x-axis)* when assuming a range of percent time active running a projected Llama3.1-8B model for 3 years. We factor idle power in $C_{operational}$ calculations, and demarcate increasing percent time active as progressively darker colors in the graph.

for the 1.5 TB and 12 TB configurations. This is because Cerebras has 4.6-9.3× higher performance (tokens/sec).

We further examine inference of a 70B model in Figure 8ii. Since 4 WSE-3 wafers are needed to serve the Llama-70B model instead of one, we observe an increase in total carbon for CS-3 configurations. This makes it such that H100 systems are more optimal from a total carbon perspective. However, if we examine carbon efficiency as $tCDP^{-1}$, we find an interesting trade-off where the most optimal design depends on the usage of the machine. After generating $O(10^{11})$ tokens, the CS-3 1.5 TB system has 9.58× better tCDP than 1 DGX box, whereas it has 0.98× the $tCDP^{-1}$ compared to 10 DGX boxes.

5.2.2 Varying Carbon Intensity. In the scenario where carbon intensity of use (CI_{use}) varies over the course of a day, the relative cross-over points would stay consistent.

For example, Figures 8 and 9 would see a shift along the axes with varying carbon intensity – i.e. if CI_{use} is lower, operational carbon and therefore total carbon would decrease, causing both curves to shift downwards and a shift to the right in the cross-over point. We note that in Figures 8 and 9 we assume both systems are running on the same grid, and therefore fluctuating carbon intensity would not change the relative conclusions of this section.

5.2.3 Factoring in Idle Power. To capture a more realistic view of system usage, we subsequently incorporate idle power into our analysis. We assume a lifetime of 3 years, and sweep the percentage of time our system is active over the course of those 3 years, assuming the system sits idle for the remaining time. We factor idle power into our operational carbon calculation using the following formula:

$$C_{operational} = (time_{active} \times power_{active} \times CI_{use}) + (time_{idle} \times power_{idle} \times CI_{use})$$

Our resulting analysis can be seen in Figure 9. As shown, we plot both total carbon and $tCDP^{-1}$ when sweeping the system's percent active time from 10-100%. Each vertical line represents the same amount of work done (i.e., tokens generated). We find that the break-even point of total carbon falls between 30-40% active use of Cerebras. That is, if Cerebras is used <30% of the time, an 8 or 10 DGX box system has lower total carbon. This is due to the high idle power of Cerebras as compared to H100 systems, as shown in Section 4. However, if the active time of the Cerebras system is high (e.g., >40%), Cerebras could see benefits from a total carbon

perspective when for a given number of tokens, the total carbon emitted is less for Cerebras than 8 or 10 DGX boxes.

We additionally examine the resulting $tCDP^{-1}$ while accounting for idle power. We find that from a carbon efficiency perspective, if the amount of work desired is such that an H100 DGX system needs to be used for >20% of the time to produce the given number of tokens, then given our assumptions in Section 5.2 and Section 6 it could be better to use Cerebras. This holds true for various DGX box configurations and results from the superior performance of Cerebras, which can generate roughly 2430 tokens/second. We further observe that if a Cerebras system is active >40% of the time, and given the aforementioned assumptions, CS-3 is at least 1.54× and 1.31× more carbon-efficient for Llama3.1-8B execution than 8 and 10 DGX boxes, respectively. Thus, our results demonstrate there is a distinct tradeoff when it comes to finding the optimal hardware for carbon efficiency, depending on the amount of work done and time the system is active. We find that wafer-scale is a promising architecture that we expect to have carbon efficiency benefits over existing GPU architectures when manufactured at high yield, *if* such systems are heavily used.

6 Limitations and Future Work

Comparing wafer-scale architectures to traditional GPUs is a challenging task. In this section, we highlight potential limitations of this study and clarify the assumptions underlying our analysis:

(1) We acknowledge yield calculations can be highly varied, depending on how defect densities are modeled. We therefore generalize our yield analysis beyond Cerebras specifically in order to capture the impact of other wafer-scale systems, as shown in Figure 4.

(2) While flash storage is an important factor to consider in embodied carbon [24, 31], we do not include flash storage in our calculations here in attempts to provide equal comparison between Cerebras and H100 systems. Since the embodied carbon of GPU architectures does not typically include for example CPU storage, we do not include the flash component of Cerebras Memory-X systems. This way we standardize the comparison of embodied carbon between the two systems by focusing on SRAM and DRAM. We note that when including flash storage in the Cerebras embodied carbon calculation, the maximum memory configuration of CS-3 has 11× higher embodied carbon due to 1500 TB flash, whereas the minimum memory configuration does not include flash.

(3) The reported FLOPS utilization measured on our Cerebras system is very low (<20%), yet power measurements suggest the hardware is being utilized 4300 W above idle power. We therefore suspect the reported utilization may not accurately reflect the typical definition of Mean Flops Utilization (MFU). Improved telemetry in future work would help this effort.

(4) For our operational carbon calculations, we assume an upper bound of power measurements for each system. For Cerebras, we utilize an upper bound on inference power by plugging in our measured training power, assuming similar utilization. In a similar manner, we use an upper bound for H100 power through TDP values. However, we acknowledge each system does not run continuously at maximum power. To tackle this, we incorporate idle power into our analysis in Figure 9. In future work, we hope to conduct more detailed power measurements of both systems and include sweeps of system utilization levels.

Furthermore, while we use CS-3 and H100 configurations as case studies here, it is important to note that specific configurations could change in future generations. We note that future work could expand our analysis to other workloads beyond LLMs as well. Our work sets the foundation for further exploration to design more carbon-aware wafer-scale systems for AI workloads across hardware lifetime.

7 Conclusion

Throughout this work, we aim to compare wafer-scale architectures versus traditional GPUs, grounding our analysis in Cerebras CS-3 and H100 DGX systems. We find that there exists a rich design space depending on model architecture and hardware configuration for determining a carbon-efficient and high performing optimal design. We hope this work inspires future optimization strategies to trade-off embodied carbon benefits of traditional GPU architectures versus energy efficiency of wafer-scale architectures.

8 Acknowledgments

The authors at Harvard University are supported by the National Artificial Intelligence Research Resource (NAIRR) Pilot and the Neocortex system supported by the National Science Foundation (award NSF-OAC 2005597) at the Pittsburgh Supercomputing Center, a joint computational research center with Carnegie Mellon University and the University of Pittsburgh.

References

[1] 2024. Cerebras CS-3: the world's fastest and most scalable AI accelerator - Cerebras. (2024). https://www.cerebras.ai/blog/cerebras-cs3

[2] 2024. Cerebras Wafer-Scale Cluster Data Sheet. (2024). https://8968533.fs1.hubspotusercontent-na1.net/hubfs/8968533/Cerebras%20Wafer%20Scale%20Cluster%20datasheet%20-%20final.pdf

[3] 2024. WSE-3 Data Sheet. (2024). https://8968533.fs1.hubspotusercontent-na1.net/hubfs/8968533/Datasheets/WSE-3%20Datasheet.pdf

[4] 2025. Cerebras Launches World's Fastest Inference for Meta Llama 4. (2025). https://www.cerebras.ai/press-release/llama4PR

[5] 2025. Llama 3.1 8B: API Provider Benchmarking Analysis. (2025). https://artificialanalysis.ai/models/llama-3-1-instruc-8b/providers

[6] 2025. Llama 3.3 70B: API Provider Benchmarking Analysis. (2025). https://artificialanalysis.ai/models/llama-3-3-instruct-70b/providers

[7] 2025. NVIDIA H100 Tensor Core GPU Data Sheet. (2025). https://www.nvidia.com/en-us/data-center/h100/

[8] Bilge Acun, Benjamin Lee, Fiodar Kazhamiaka, Kiwan Maeng, Udit Gupta, Manoj Chakkaravarthy, David Brooks, and Carole-Jean Wu. 2023. Carbon Explorer: A Holistic Framework for Designing Carbon Aware Datacenters. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2* (Vancouver, BC, Canada) *(ASPLOS 2023)*. Association for Computing Machinery, New York, NY, USA, 118–132. doi:10.1145/3575693.3575754

[9] Nystrom N.A. Buitrago P.A. 2021. Neocortex and Bridges-2: A High Performance AI+HPC Ecosystem for Science, Discovery, and Societal Good. *Nesmachnow S., Castro H., Tchernykh A. (eds) High Performance Computing. CARLA 2020. Communications in Computer and Information Science, vol 1327. Springer, Cham.* (2021). https://doi.org/10.1007/978-3-030-68035-0_15

[10] Cerebras. 2025. 100x Defect Tolerance: How Cerebras Solved the Yield Problem - Cerebras. https://www.cerebras.ai/blog/100x-defect-tolerance-how-cerebras-solved-the-yield-problem

[11] Niladrish Chatterjee, Mike O'Connor, Donghyuk Lee, Daniel R. Johnson, Stephen W. Keckler, Minsoo Rhu, and William J. Dally. 2017. Architecting an Energy-Efficient DRAM System for GPUs. In *2017 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. 73–84. doi:10.1109/HPCA.2017.58

[12] J.A. Cunningham. 1990. The use and evaluation of yield models in integrated circuit manufacturing. *IEEE Transactions on Semiconductor Manufacturing* 3, 2 (1990), 60–71. doi:10.1109/66.53188

[13] Mariam Elgamal, Doug Carmean, Elnaz Ansari, Okay Zed, Ramesh Peri, Srilatha Manne, Udit Gupta, Gu-Yeon Wei, David Brooks, Gage Hills, and Carole-Jean Wu. 2023. Carbon-Efficient Design Optimization for Computing Systems. In *Proceedings of the 2nd Workshop on Sustainable Computer Systems* (Boston, MA, USA) *(HotCarbon '23)*. Association for Computing Machinery, New York, NY, USA, Article 16, 7 pages. doi:10.1145/3604930.3605712

[14] Mariam Elgamal, Doug Carmean, Elnaz Ansari, Okay Zed, Ramesh Peri, Srilatha Manne, Udit Gupta, Gu-Yeon Wei, David Brooks, Gage Hills, and Carole-Jean Wu. 2025. CORDOBA: Carbon-Efficient Optimization Framework for Computing Systems. In *2025 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. 1289–1303. doi:10.1109/HPCA61900.2025.00098

[15] Farbin Fayza, Satyavolu Papa Rao, Darius Bunandar, Udit Gupta, and Ajay Joshi. 2024. Photonics for Sustainable Computing. arXiv:2401.05121 [cs.ET] https://arxiv.org/abs/2401.05121

[16] Danielle Grey-Stewart, Mariam Elgamal, David Kong, Georgios Kyrizaidis, Jalil Morris, and Gage Hills. 2025. Quantifying Trade-Offs in Power, Performance, Area, and Carbon Footprint of Future Computing Systems. In *2025 Design, Automation and Test in Europe Conference (DATE '25)*.

[17] Udit Gupta, Mariam Elgamal, Gage Hills, Gu-Yeon Wei, Hsien-Hsin S. Lee, David Brooks, and Carole-Jean Wu. 2022. ACT: Designing Sustainable Computer Systems with an Architectural Carbon Modeling Tool. In *Proceedings of the 49th Annual International Symposium on Computer Architecture*. Association for Computing Machinery, New York, NY, USA. doi:10.1145/3470496.3527408

[18] imec. 2025. imec.netzero web application. https://netzero.imec-int.com

[19] Yueying Li, Zhanqiu Hu, Esha Choukse, Rodrigo Fonseca, G. Edward Suh, and Udit Gupta. 2025. EcoServe: Designing Carbon-Aware AI Inference Systems. arXiv:2502.05043 [cs.DC] https://arxiv.org/abs/2502.05043

[20] Sean Lie. 2022. Cerebras Architecture Deep Dive: First Look Inside the HW/SW Co-Design for Deep Learning. (2022). https://hc34.hotchips.org/assets/program/conference/day2/Machine%20Learning/HC2022_Cerebras_Final_v02.pdf

[21] Sean Lie. 2023. Cerebras Architecture Deep Dive: First Look Inside the Hardware/Software Co-Design for Deep Learning. *IEEE Micro* 43, 3 (2023), 18–30. doi:10.1109/MM.2023.3256384

[22] Sean Lie. 2024. Inside the Cerebras Wafer-Scale Cluster. *IEEE Micro* 44, 3 (2024), 49–57. doi:10.1109/MM.2024.3386628

[23] Sean Lie. 2024. Wafer-Scale AI: GPU Impossible Performance. (2024). https://hc2024.hotchips.org/assets/program/conference/day2/72_HC2024.Cerebras.Sean.v03.final.pdf

[24] Sara Mcallister, Fiodar Kazhamiaka, Daniel S. Berger, Rodrigo Fonseca, Kali Frost, Aaron Ogus, Maneesh Sah, Ricardo Bianchini, George Amvrosiadis, Nathan Beckmann, and Gregory R. Ganger. 2025. A Call for Research on Storage Emissions. *SIGENERGY Energy Inform. Rev.* 4, 5 (April 2025), 67–75. doi:10.1145/3727200.3727211

[25] Jack F. McDonald, Edwin H. Rogers, Kenneth Rose, and Andrew J. Steckl. 1984. The trials of wafer-scale integration: Although major technical problems have been overcome since WSI was first tried in the 1960s, commercial companies can't yet make it fly. *IEEE Spectrum* 21, 10 (1984), 32–39. doi:10.1109/MSPEC.1984.6370295

[26] B.T. Murphy. 1964. Cost-size optima of monolithic integrated circuits. *Proc. IEEE* 52, 12 (1964), 1537–1545. doi:10.1109/PROC.1964.3442

[27] Sophia Nguyen, Beihao Zhou, Yi Ding, and Sihang Liu. 2025. Towards Sustainable Large Language Model Serving. *SIGENERGY Energy Inform. Rev.* 4, 5 (April 2025), 134–140. doi:10.1145/3727200.3727220

[28] Saptadeep Pal, Daniel Petrisko, Matthew Tomei, Puneet Gupta, Subramanian S. Iyer, and Rakesh Kumar. 2019. Architecting Waferscale Processors - A GPU Case Study. In *2019 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. 250–263. doi:10.1109/HPCA.2019.00042

[29] Raghu Prabhakar, Ram Sivaramakrishnan, Darshan Gandhi, Yun Du, Mingran Wang, Xiangyu Song, Kejie Zhang, Tianren Gao, Angela Wang, Xiaoyan Li, Yongning Sheng, Joshua Brot, Denis Sokolov, Apurv Vivek, Calvin Leung, Arjun Sabnis, Jiayu Bai, Tuowen Zhao, Mark Gottscho, David Jackson, Mark Luttrell, Manish K. Shah, Zhengyu Chen, Kaizhao Liang, Swayambhoo Jain, Urmish Thakker, Dawei Huang, Sumti Jairath, Kevin J. Brown, and Kunle Olukotun. 2024. SambaNova SN40L: Scaling the AI Memory Wall with Dataflow and Composition of Experts. In *2024 57th IEEE/ACM International Symposium on Microarchitecture (MICRO)*. IEEE, 1353–1366. doi:10.1109/micro61859.2024.00100

[30] L-Å Ragnarsson, M. Garcia Bardon, P. Wuytens, G. Mirabelli, D. Jang, G. Willems, A. Mallik, A. Spessot, J. Ryckaert, and B. Parvais. 2022. Environmental Impact of CMOS Logic Technologies. In *2022 6th IEEE Electron Devices Technology & Manufacturing Conference (EDTM)*. 82–84. doi:10.1109/EDTM53872.2022.9798208

[31] Varsha Rao and Andrew A. Chien. 2025. Understanding the Operational Carbon Footprint of Storage Reliability and Management. *SIGENERGY Energy Inform. Rev.* 4, 5 (April 2025), 180–187. doi:10.1145/3727200.3727227

[32] Arman Shehabi, Alex Hubbard, Alex Newkirk, Nuoa Lei, Md Abu Bakkar Siddik, Billie Holecek, Jonathan Koomey, Eric Masanet, and Dale Sartor. 2024. 2024 United States Data Center Energy Usage Report. Lawrence Berkeley National Laboratory, Berkeley, California, LBNL–2001637.

[33] Sean Lie Stewart Hall, Rob Schreiber. 2023. Training Giant Neural Networks Using Weight Streaming on Cerebras Wafer-Scale Clusters. (2023). https://8968533.fs1.hubspotusercontent-na1.net/hubfs/8968533/Virtual%20Booth%20Docs/CS%20Weight%20Streaming%20White%20Paper.pdf

[34] Jovan Stojkovic, Chaojie Zhang, Íñigo Goiri, Josep Torrellas, and Esha Choukse. 2025. DynamoLLM: Designing LLM Inference Clusters for Performance and Energy Efficiency. In *2025 IEEE International Symposium on High Performance Computer Architecture (HPCA)*. 1348–1362. doi:10.1109/HPCA61900.2025.00102

[35] Chetan Choppali Sudarshan, Nikhil Matkar, Sarma Vrudhula, Sachin S. Sapatnekar, and Vidya A. Chhabria. 2024. ECO-CHIP: Estimation of Carbon Footprint of Chiplet-based Architectures for Sustainable VLSI. In *2024 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. 671–685. doi:10.1109/HPCA57654.2024.00058

[36] Jennifer Switzer, Gabriel Marcano, Ryan Kastner, and Pat Pannuto. 2023. Junkyard Computing: Repurposing Discarded Smartphones to Minimize Carbon. In *Proceedings of the 28th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2* (Vancouver, BC, Canada) *(ASPLOS 2023)*. Association for Computing Machinery, New York, NY, USA, 400–412. doi:10.1145/3575693.3575710

[37] Yujie Zhao, Yang (Katie) Zhao, Cheng Wan, and Yingyan (Celine) Lin. 2024. 3D-Carbon: An Analytical Carbon Modeling Tool for 3D and 2.5D Integrated Circuits. In *Proceedings of the 61st ACM/IEEE Design Automation Conference* (San Francisco, CA, USA) *(DAC '24)*. Association for Computing Machinery, New York, NY, USA, Article 178, 6 pages. doi:10.1145/3649329.3658482