The Sunk Carbon Fallacy: Rethinking Carbon Footprint Metrics for Effective Carbon-Aware Scheduling

Noman Bashir
MIT

Varun Gohil
MIT

Anagha Belavadi
MIT

Mohammad Shahrad
University of British Columbia

David Irwin
University of Massachusetts Amherst

Elsa Olivetti
MIT

Christina Delimitrou
MIT

ABSTRACT

The rapid increase in computing demand and corresponding energy consumption have focused attention on computing's impact on the climate and sustainability. Prior work proposes metrics that quantify computing's carbon footprint across several lifecycle phases, including its supply chain, operation, and end-of-life. Industry uses these metrics to optimize the carbon footprint of manufacturing hardware and running computing applications. Unfortunately, prior work on optimizing datacenters' carbon footprint often succumbs to the *sunk cost fallacy* by considering embodied carbon emissions (a sunk cost) when making operational decisions (i.e., job scheduling and placement), which leads to operational decisions that do not always reduce the total carbon footprint.

In this paper, we evaluate carbon-aware job scheduling and placement on a given set of servers for several carbon accounting metrics. Our analysis reveals state-of-the-art carbon accounting metrics that include embodied carbon emissions when making operational decisions can increase the total carbon footprint of executing a set of jobs. We study the factors that affect the added carbon cost of such suboptimal decision-making. We then use a real-world case study from a datacenter to demonstrate how the sunk carbon fallacy manifests itself in practice. Finally, we discuss the implications of our findings in better guiding effective carbon-aware scheduling in on-premise and cloud datacenters.

SoCC '24, November 20–22, 2024, Redmond, WA, USA

ACM ISBN 979-8-4007-1286-9/24/11.
https://doi.org/10.1145/3698038.3698542

CCS CONCEPTS

• **Hardware → Impact on the environment**; **Emerging tools and methodologies**.

KEYWORDS

Sustainable computing, operational and embodied carbon footprint, sustainability, metrics, datacenters, scheduling.

ACM Reference Format:
Noman Bashir, Varun Gohil, Anagha Belavadi, Mohammad Shahrad, David Irwin, Elsa Olivetti, and Christina Delimitrou. 2024. The Sunk Carbon Fallacy: Rethinking Carbon Footprint Metrics for Effective Carbon-Aware Scheduling. In *ACM Symposium on Cloud Computing (SoCC '24), November 20–22, 2024, Redmond, WA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3698038.3698542

1 INTRODUCTION

Computing demand has skyrocketed over recent decades, with no signs of slowing [18]. This demand is likely accelerating due to the rise of computationally intensive generative AI tools, such as ChatGPT [13] and GitHub Copilot [45], which promise to unlock a wide range of innovative applications. However, as marginal improvements in computing's energy efficiency shrink due to the slowdown in process scaling [27, 54], the growing demand for computing power is expected to drive a proportional increase in energy consumption. This rising energy footprint has sparked significant concerns about computing's impact on climate and sustainability. Fortunately, awareness of the need to improve computing's sustainability is increasing [8, 46, 60], with coordinated efforts from both industry and academia to mitigate its environmental impact [9, 41, 55, 56, 60].

Recent efforts to improve computing's sustainability have focused on quantifying and optimizing its carbon footprint across all lifecycle stages, from chip design and manufacturing [1, 24] to system operations [25, 28, 43] and e-waste

management [49]. The Greenhouse Gas (GHG) Protocol [59] highlights two key emission types: Scope 2 covers emissions from electricity use in datacenters (operational emissions); Scope 3 includes emissions from chip manufacturing, supply chains, and e-waste management (embodied emissions).

Previous work on computing's carbon footprint has used various metrics, typically based on operational emissions alone or a weighted combination of operational and embodied emissions. A common approach aggregates a job's operational emissions with a portion of the server's embodied emissions, distributing the server's embodied emissions across jobs based on their resource usage and duration. Notable examples include the Software Carbon Intensity (SCI) introduced by the Green Software Foundation [20], Computational Carbon Intensity [49], and Sustainability Cost Rate [21]. Though these metrics use different terms, they follow the same core principle: a job's carbon footprint is the sum of its share of the hardware's embodied emissions and the operational emissions generated during its execution.

In this paper, we focus on carbon-aware workload scheduling and job placement on datacenter servers. While embodied carbon-based metrics like SCI are often proposed to guide operational decisions, such as scheduling and job placement, we argue that scheduling and procurement operate on different timescales and should be optimized independently. Scheduling determines which servers handle specific jobs and should focus on minimizing the operational carbon footprint of active servers. In contrast, procurement decisions–such as which servers to purchase and when to replace them–affect the embodied carbon footprint from hardware manufacturing, which cannot be influenced once a job is being scheduled. These processes are distinct: scheduling occurs continuously as jobs are assigned, while procurement decisions are made periodically based on hardware lifecycles.

Importantly, metrics like SCI, which incorporate lifecycle emissions, typically account only for the emissions of servers running jobs, ignoring the embodied carbon of idle servers. This oversight can lead to unintended consequences when optimizing for SCI-like metrics in job scheduling, paradoxically increasing a datacenter's overall carbon footprint by neglecting the broader carbon impact of idle hardware. We show that focusing solely on SCI-like metrics in scheduling may undermine the goal of minimizing a datacenter's total carbon footprint, underscoring the need for separate, independent optimization of scheduling and procurement.

The suboptimal outcomes of carbon-aware scheduling based on SCI-like metrics stem from a cognitive bias known as the *sunk cost fallacy*. According to the *principle of bygones*, rooted in economic theory's principle of separability, decisions should focus solely on future possibilities without being influenced by past expenditures or irreversible events [17]. Applied to datacenter operations, scheduling and job placement decisions should prioritize the current operational context, disregarding embodied emissions that have already occurred. The embodied emissions are fixed at procurement and cannot be changed through operational decisions; operators should prioritize operational carbon.

Ignoring sunk costs is intuitive and supported by prior research [22, 39, 48, 57]. However, recent efforts to develop metrics that optimize computing's lifecycle carbon footprint have unintentionally introduced a *sunk carbon fallacy*, a variant of the sunk cost fallacy applied to carbon. These metrics conflate procurement and operation by incorporating embodied emissions into real-time scheduling decisions. As our example in Section 3.2 shows, using SCI as a scheduling metric can paradoxically increase a datacenter's overall carbon footprint, highlighting the need to optimize scheduling and procurement independently for true carbon efficiency.

The extent to which minimizing a datacenter's total carbon footprint diverges from minimizing the sum of job-level lifecycle carbon using metrics like SCI depends on several infrastructure characteristics. One key factor is the heterogeneity in server performance relative to their operational and embodied carbon footprints. In a datacenter with homogeneous servers—where performance is similar across all servers—incorporating embodied carbon into a scheduling metric like SCI would not significantly affect the overall system-level carbon footprint. However, real-world datacenters are often heterogeneous, differing in hardware age (e.g., new vs. old) and type (e.g., CPU vs. GPU). For example, older servers generally have lower embodied carbon due to earlier manufacturing but have higher operational carbon emissions than newer, energy-efficient servers, as shown in Figure 1. Moreover, GPUs are ideal for compute-intensive tasks; CPUs may perform better per unit of carbon for specific tasks [2].

Our work focuses on CPU heterogeneity, which is significant enough to show that applying a one-size-fits-all metric like SCI, which includes embodied carbon, can distort scheduling decisions and increase the overall carbon footprint. Another critical factor is datacenter utilization. When utilization is either very high or very low — where all servers are in use or none are — the choice of scheduling metric has little impact on the total carbon footprint. However, at intermediate utilization levels, common in many datacenters, metrics like SCI can lead to inefficient scheduling, thereby increasing the total carbon footprint. In Section 3.2, we examine this discrepancy and evaluate the impact of infrastructure factors on a datacenter's carbon footprint using concrete examples.

In showing how the *sunk cost fallacy* manifests in carbon-aware scheduling, we make the following contributions:
1 – We show that metrics incorporating both embodied and operational carbon emissions, while seemingly comprehensive, can lead to suboptimal scheduling decisions. These metrics may paradoxically increase a datacenter's overall carbon

footprint, contradicting their intended goal. We explore key factors such as datacenter utilization, operational carbon intensity, and embodied carbon amortization approaches that exacerbate these suboptimal outcomes.
2 – We evaluate three metrics, including those that prioritize operational emissions or account for infrastructure-wide embodied carbon better than `SCI`. Through a real-world case study of an on-premise datacenter, we demonstrate that under realistic workload conditions, focusing on operational carbon emissions results in more carbon-efficient scheduling and a reduced total carbon footprint.
3 – We provide practical guidelines for datacenter operators and users to avoid the sunk carbon fallacy. Our recommendations emphasize selecting metrics that accurately reflect carbon costs relevant to operational decisions, enabling optimization for a lower overall carbon footprint.

2 BACKGROUND AND MOTIVATION

Prior work on sustainable computing. There has been extensive research on the environmental impact of computing [46] and on defining what sustainable computing entails [11, 60]. Prior work has also analyzed various carbon accounting frameworks within computing, highlighting the challenges of accurately assessing its carbon footprint [10, 30], particularly regarding the error-prone nature of embodied carbon values [10] and operational carbon intensity estimates [30]. Recent studies have focused on quantifying both operational and embodied carbon and their tradeoffs to inform architectural design aimed at reducing servers' overall lifecycle carbon footprint. Prior work has explored the potential benefits and limitations of spatiotemporal workload scheduling for reducing carbon emissions [47]. In parallel, researchers have developed algorithms for carbon-aware workload shifting and built system support for such algorithms [23, 25, 43, 52]. Despite the advances, the real-world adoption of carbon-aware optimizations remains limited, with only one notable example of carbon-aware workload shifting implemented by hyperscalers [41].
Metrics for sustainable computing. Recent work has explored various metrics to quantify and optimize computing's carbon footprint. Gandhi et al. [21] propose sustainability metrics for datacenters, including the amortized sustainability cost metric, which attributes both operational and embodied carbon to a job. Switzer et al. [49] address the end-of-life problem for computing hardware and propose the computational carbon intensity (CCI) metric, which aids in making decisions about component replacement and end-of-life management. The software industry has also focused on promoting green software development, with initiatives from the Green Software Foundation (GSF) [20], which introduced the Software Carbon Intensity (SCI) metric to help developers quantify and reduce software's carbon footprint.
Limitations and research gaps. Previous work on carbon accounting has introduced various metrics to reduce computing's carbon footprint, sparking debate about their usefulness and effectiveness [14, 15, 19, 42]. Despite the critical nature of the problem, little research has focused on analyzing the incentives each metric provides and the outcomes they produce. Recent studies suggest that creating a single metric that is simple, accurate, precise, and offers the right incentives for optimizing decision-making across computing's entire lifecycle may not be feasible [50]. Moreover, evaluating all possible metric combinations presents a significant challenge. The total lifecycle carbon footprint includes the embodied carbon of all servers, operational carbon from idle servers, and emissions from active servers running workloads. Procurement decisions and job scheduling affect this footprint, but they operate on different timescales: seconds to days for scheduling and months to years for procurement. This work focuses on carbon-aware workload scheduling in public cloud and enterprise datacenters, targeting reductions in the carbon footprint added during this lifecycle stage.

3 THE SUNK CARBON FALLACY

This section shows how state-of-the-art carbon accounting metrics fall prey to the *sunk carbon fallacy*, outlines factors contributing to suboptimal decision-making, and examines metrics that yield better carbon-aware scheduling outcomes.
Setup. Carbon-aware scheduling assigns jobs to available servers to minimize the total carbon footprint of executing those jobs. In our example, we assume the following setup:
– **The scheduler** aims to place jobs on servers to minimize the total carbon footprint without knowledge of future job arrivals or characteristics, making instantaneous placement decisions—similar to production schedulers like Borg [7, 53].
– **Jobs** performance characteristics and energy usage on given servers are known through profiling or public databases like MLPerf [32] and OpenBenchmarking Suite [38].
– **Servers** are not power-proportional, consuming significant power even at 0% utilization [6, 29], often exceeding 30% of peak usage. However, the idle power for processing components is much lower. While individual servers may be fully utilized, datacenter-level utilization typically ranges between 30% to 60%, even in state-of-the-art facilities [53].
– **Energy and carbon footprint estimates** for servers depend on components like power supplies, hard drives, memory, and chassis. We use data from MIT's Bates Research and Engineering Center [33] and the hydro-powered Massachusetts Green High Performance Computing Center (MGHPCC) [34] that provides processor information.

Embodied carbon for processors is estimated using the PAIA integrated circuit module [37], based on factors like technology node (e.g., 7nm, 28nm), CPU package area, die size, and fabrication location. Technology node and

CPU package area data are sourced from official Intel and AMD websites, while die sizes are gathered from TechPowerUp [51], CPU-World [16], X86 CPU's Guide [35], and WikiChip [58], with cross-verification for consistency. We use a carbon intensity of 495g.CO_2/kWh for AMD processors fabricated in Taiwan [36] and 357g.CO_2/kWh for Intel processors fabricated in Hillsboro, Oregon [31].

For operational carbon estimates, we assume servers consume their rated Thermal Design Power (TDP) at 100% utilization, with a linear power increase between idle and full load. We assume datacenter is in Sweden, with a carbon intensity of 20g.CO_2/kWh [31], and vary this intensity for analysis in both embodied- and operational-dominant regions.

– **Performance Benchmarks.** Processor performance scores are based on three benchmarks: Multithread Ratings by PassMark [40], HEPScore [26], and SPEC CPU2017 Floating Point Speed [44]. Not all benchmarks profile every processor, which narrows the set of processors in our analysis.

3.1 Carbon-Aware Scheduling Metrics

This section defines three different metrics that can be used to evaluate carbon-aware scheduling and job placement.

1 – Software Carbon Intensity (SCI) was introduced by the Green Software Foundation [20] to quantify the rate of total carbon emissions per functional unit **R**, which could be an API call, machine learning (ML) training, or AI inference. The carbon emissions for a given job consist of both operational carbon emissions (denoted O) from running the job on the server, and embodied carbon emissions (denoted M) for the functional unit. SCI is defined as:

$$\texttt{SCI} = (\texttt{O} + \texttt{M}) \text{ per } \texttt{R} = ((\texttt{E} * \texttt{I}) + \texttt{M}) \text{ per } \texttt{R},$$

where E is the job's energy consumption (in kilowatt-hours) over a given time window, including a portion of the server's idle and dynamic power usage. I is the carbon intensity of electricity, measured in grams of CO2 equivalent per kilowatt-hour (g.CO_2/kWh), for the region where the server operates. SCI accounts only for the embodied carbon (M) of the active server running the job, computed as:

$$\texttt{M} = \texttt{TE} \times {}^{\texttt{T} \times \texttt{RR}}/_{\texttt{EL} \times \texttt{TR}}. \quad (1)$$

Here, TE is the total embodied emissions, EL is the server's expected lifespan, and TR represents the server's total resources. T denotes the time duration, and RR is the resource reserved for the job (see SCI specifications for further details [20]).

2 – Total Software Carbon Intensity (tSCI) extends **SCI** by incorporating the embodied carbon emissions of the entire infrastructure, aiming for a more accurate representation of total emissions. Instead of accounting only for the server running the job, tSCI distributes a portion of the total embodied emissions across all jobs, including idle infrastructure.

To extend SCI, we add a fraction of the infrastructure-level embodied carbon based on the resources reserved and the job's allotted time, with a total embodied carbon (tM) of $\texttt{tM} = \texttt{M} + \texttt{M}_{\text{idle-infra}}$, where $\texttt{M}_{\text{idle-infra}}$ is the embodied carbon of idle servers, calculated using the same method as M in Equation 1. Each idle server's embodied carbon is proportionally assigned to the job. Similarly, to account for the operational carbon from idle servers, tO is computed as $\texttt{tO} = \texttt{O} + \texttt{O}_{\text{idle-infra}}$. The total software carbon intensity is then:

$$\texttt{tSCI} = (\texttt{tO} + \texttt{tM}) \text{ per } \texttt{R}.$$

Table 1: Specifications of servers in our example.

	S_A	S_B
Processor	Xeon E-2286G	Xeon Gold 6538N
Release Date	05/29/2019	12/14/2023
PassMark Score	14020	44895
TDP (W)	90	205
Technology Node	14nm	10nm
Embodied Carbon (Kg.CO2)	8.04	101.89

To illustrate this, consider a datacenter with two servers, A and B, with embodied carbon values of 400g.CO_2 and 50g.CO_2, and expected lifetimes of 10 and 5 years, respectively. Server A has 40 cores, and server B has 10 cores. Suppose job J_1, which runs for one year using 10 cores, is scheduled on server B, while job J_2, also using 10 cores, runs on server A. The embodied carbon attributed to J_1 is:

$$\texttt{tM} = 10\text{g.CO}_2 + \underbrace{\frac{400\text{g.CO}_2 \times 1yr}{10yrs}}_{\text{time fraction}} \times \underbrace{\frac{30cores}{40cores}}_{\text{idle fraction}} \times \underbrace{\frac{10cores}{20cores}}_{\text{usage fraction}},$$
$$= 25\text{g.CO}_2.$$

The time, idle, and usage fractions amortize the embodied carbon of the idle infrastructure over time (1 out of 10 years), idle resources (30 out of 40 cores are idle), and usage (10 of the 20 total used cores). The operational carbon emission rate, tO, can be computed similarly, except for the time component.

3 – Operational Software Carbon Intensity (oSCI) metric ignores the embodied carbon emissions for all the servers. It makes scheduling decisions based on the operational carbon emissions of running a given job. oSCI is expressed as,

$$\texttt{oSCI} = (\texttt{E} * \texttt{I}) \text{ per } \texttt{R}.$$

This metric can include a portion of the base power from the idle servers to incentivize turning off servers when they are idle. However, for the current purpose, we keep it simple and only account for the energy used by the job's server.

Computing SCI, tSCI, and oSCI in Practice presents varying levels of complexity. oSCI, a subset of the other metrics, is the simplest to calculate as job operating power can be estimated through offline profiling. SCI, however, requires embodied carbon estimates for all servers in a datacenter, which can be difficult to obtain and often have significant uncertainty [3, 12, 37]. This uncertainty can propagate unpredictably, affecting scheduling outcomes. Calculating tSCI

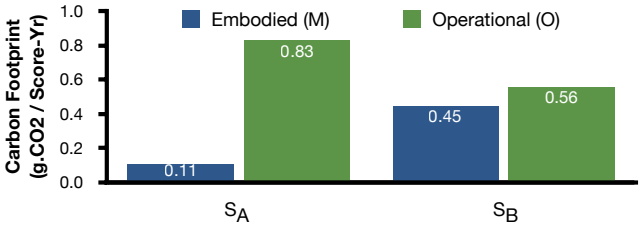

Figure 1: The normalized embodied and operational carbon footprint (g.CO_2) per Score-Yr for a datacenter in Sweden with electricity's carbon intensity of 14g.CO_2/kWh [31]. The servers have a lifetime of 5 years.

and tracking it over time is even more complex, requiring comprehensive datacenter-level information, including all hardware, active jobs, resource reservations, and runtime expectations. The idle fraction of infrastructure varies as jobs arrive and leave, resulting in a time-varying tSCI. While cloud operators have access to this data, calculating tSCI requires sophisticated infrastructure and precise attribution, both costly and carbon-intensive. Public cloud users generally lack access to such data, making it impractical for them to compute their carbon footprint. Thus, we do not expect tSCI to be widely used in practice, and instead, we show that oSCI can achieve similar scheduling outcomes with less complexity. Finally, integrating operational and embodied carbon estimates into scheduling decisions depends on the scheduler. For example, in Slurm, nodes can be assigned weights reflecting the chosen metric, such as oSCI. Slurm's energy monitoring tools can be easily modified to report operational emissions with minimal overhead.

3.2 An Illustrative Example

We first use a simple example to demonstrate the *sunk carbon fallacy*. Consider a small datacenter with two servers powered by two processors from Intel: Xeon E-2286G and Xeon Gold 6538N, referred to as S_A and S_B, respectively. Table 1 provides the detailed specifications for the two servers, including processor model, their release dates, PassMark scores, embodied carbon estimates, and TDP values.

Figure 1 shows the operational and embodied carbon emissions normalized to the PassMark score and expected lifetime of two servers in our dummy datacenter. An operational carbon value of 0.56 means that achieving a performance score of 1 for one year using S_A results in 0.56g.CO_2 of operational emissions. This example reflects a common scenario: a newer server (S_B), manufactured with 10nm technology, has 4.09× the embodied carbon footprint of an older server (S_A) using 14nm technology. However, energy efficiency gains over recent years mean S_B consumes 32.5% less energy than S_A.

1 – Analyzing Scheduling Outcomes. Table 2 presents the carbon footprint values used to choose a server for job placement. It includes the total lifecycle emissions of the datacenter during the job's execution, encompassing both the embodied carbon for all servers and the operational carbon of active servers. The server with the lowest metric is highlighted in bold and chosen to run the job. The datacenter-level carbon footprint is the sum of the embodied carbon (the sunk cost) for all servers and the operational carbon for the server running the job (the marginal or additional cost).

Table 2: Values of SCI, tSCI, oSCI for S_A and S_B for job placement in g.CO_2 per Score-Yr. We also report the total cluster carbon footprint for each metric.

Metric	Scheduling/Placement		Accounting
	S_A	S_B	Cluster Carbon Footprint
SCI	0.11 + 0.83 = **0.94**	0.45 + 0.56 = 1.01	(0.11 + 0.45) + 0.83 = 1.39
tSCI	0.94 + 0.45 = 1.39	1.01 + 0.11 = **1.12**	(0.11 + 0.45) + 0.56 = 1.12
oSCI	0.83	**0.56**	(0.11 + 0.45) + 0.56 = 1.12

As shown, when prioritizing the sum of embodied and operational emissions, the SCI metric selects a highly energy-inefficient server due to its low embodied carbon. While this decision minimizes SCI, it results in a 24.10% higher carbon footprint for the datacenter. In contrast, the placement choices of tSCI and oSCI align, leading to the minimum cluster-level emissions, as both prioritize minimizing additional emissions while achieving the desired performance.

This example illustrates the classic scenario of a new, energy-efficient server with high embodied carbon versus an old, energy-inefficient server with low embodied carbon, mainly due to the technology node difference. However, this mismatch can occur where an energy-inefficient server has a lower SCI value than a more efficient server. For instance, as shown in Table 3, the newer Xeon E-2486 server, built on a 10nm node, has a smaller embodied carbon footprint than the EPYC 9334 server. Despite energy efficiency gains and performance improvements, the EPYC 9334 server's higher embodied carbon results in a larger SCI value.

A similar situation arises between Ryzen Threadripper 5965WX and Xeon W9-3495. The Ryzen processor, built on a 5nm node, has a lower embodied carbon footprint than the Xeon processor, which uses a 10nm node, despite the latter's advanced manufacturing process. These examples demonstrate that the *sunk carbon fallacy* extends beyond the old vs. new server comparison, as even servers not intended as direct replacements can still coexist in a datacenter or cloud platform, leading to the selection of an inefficient server.

2 – Effect of Datacenter Utilization. Our example shows how variations in server characteristics lead to suboptimal scheduling. We now explore the effect of datacenter utilization on system-level carbon footprint increases when using SCI. Server S_A has 12 logical cores (6 physical cores, 2 threads per core); each logical core has an 1168 PassMark score. Server S_B has 64 logical cores (32 physical cores, 2 threads per core); each logical core has a 701 PassMark score.

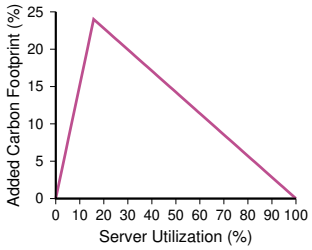

Figure 2: Utilization Impact

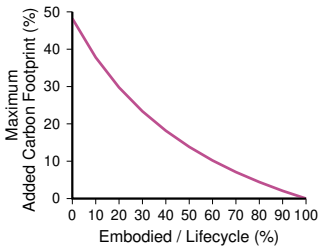

Figure 3: Carbon Impact

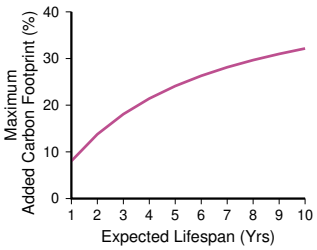

Figure 4: Lifespan Impact

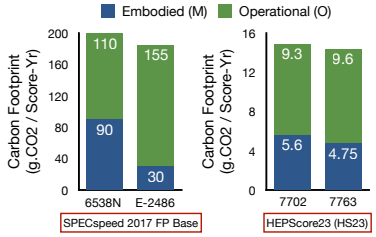

Figure 5: Benchmark Impact

Each job uses one logical core on S_A and two logical cores on S_B to achieve a performance score of 1402 (closer to the 1168 for S_A), giving us 44 cores of similar performance.

Figure 2 illustrates the increase in system-level carbon footprint when jobs are scheduled using SCI, compared to scheduling based on tSCI or oSCI. When datacenter utilization is either 0% or 100%, all metrics yield the same result. However, at intermediate utilization levels, the choice of server becomes important. The peak discrepancy occurs when only 12 cores are needed to run the jobs (at 27.3% utilization). The exact peak and the utilization level at which it occurs will vary depending on the server set, their base power values, and the scheduling granularity. In Section 4, we present similar results for our case study.

3 – Effect of Operational Carbon Intensity. In our setup, embodied carbon accounts for 11.7% and 44.5% of the lifecycle emissions for S_A and S_B, respectively, with an average of 28.1% across servers. To analyze the impact of operational carbon intensity, we scale the normalized operational carbon to make embodied carbon account for 10% to 90% of lifecycle emissions. Figure 3 shows the maximum added carbon footprint due to the sunk carbon fallacy as embodied carbon accounts for a higher share of lifecycle emissions. At 0%, where only operational efficiency matters, using S_A results in a 48% increase in system-level carbon. Conversely, at 100%, operational carbon is zero, and server choice is irrelevant.

Despite placing the datacenter in Sweden – a region with one of the world's lowest carbon intensities – operational emissions still dominate because our embodied carbon estimates focus only on the processor, which is a small portion of the server-level carbon footprint. In contrast, the processor's TDP accounts for most of the server's power and operational carbon footprint. If server-level embodied carbon values were used, the carbon intensity at which embodied carbon makes a given %age of lifecycle emissions would be higher.

4 – Effect of Server's Expected Lifetime. The expected lifespan of servers has a similar impact on the added carbon footprint at the system level. Figure 4 shows the maximum added carbon footprint at the system-level as the server's embodied carbon is amortized over a longer period. As the expected lifespan increases, the amortized embodied carbon per year decreases, and its fraction of the lifecycle carbon footprint decreases. As shown in Figure 3, lower embodied values result in a higher system-level carbon footprint under SCI, magnifying the impact of the *sunk carbon fallacy*.

Table 3: Additional scenarios of sunk carbon fallacy. Values of carbon emissions are in $\mathtt{g.CO_2}$ per Score-Yr.

Server Pairs		Additional Details
Xeon E-2486	EPYC 9334	New Xeon server (12/14/2023, 10nm)
0.08 + 0.47 = **0.55**	0.23 + 0.39 = 0.62	vs. old EPYC server (11/10/2022, 5nm).
Ryzen 5965WX	Xeon W9-3495	Older Ryzen server (03/08/2022, 5nm)
0.15 + 0.51 = **0.66**	0.25 + 0.46 = 0.71	vs. New Xeon server (02/15/2023, 10nm).

5 – Effect of Performance Metric. Our results thus far have used PassMark scores. However, our observation is agnostic to any particular benchmarking method. Figure 5 shows that the conditions required for the *sunk carbon fallacy*, i.e., a server with low SCI is inefficient, manifest across different benchmarks. The servers we use in our examples changed, as we did not have SPEC and HS26 scores for the servers in the illustrative example. While the combination of servers that manifest the *sunk carbon fallacy* may change, the effect should be present in all performance benchmarks.

Generalization of Outcomes We now explore whether our observations hold across different hardware configurations, considering their embodied and operational carbon ratios. Assume there are N servers in a datacenter, and k servers are needed at any time. Let M_i and O_i represent the embodied and operational carbon costs of server i, and let $Z_i = M_i + O_i$ denote the total carbon emissions over the server's lifetime.

The SCI and oSCI strategies are formalized as:

$$\mathtt{SCI} = \{i \mid Z_i \text{ are the } k \text{ smallest values of } Z\},$$
$$\mathtt{oSCI} = \{i \mid O_i \text{ are the } k \text{ smallest values of } O\}.$$

If k = 0 or k = N, both strategies select the same servers. However, oSCI minimizes $\sum_{i \in \mathtt{oSCI}} O_i$, the operational carbon, which can be reduced post-purchase. In contrast, SCI might pick servers with lower lifecycle costs Z_i but higher O_i, resulting in suboptimal choices. Therefore:

$$\sum_{i \in \mathtt{oSCI}} O_i \leq \sum_{i \in \mathtt{SCI}} O_i.$$

Since total carbon emissions include both embodied and operational phases, oSCI ensures the lowest footprint across purchase and operation. While extending this example to dynamic job arrivals shows similar results, a detailed exploration of that scenario is beyond this paper's scope and will be addressed in future work.

Table 4: List of servers and their specifications for the case study. The server life is 5 years; for older than 5 years old servers embodied carbon is amortized over years since purchase. The operational carbon is for 5 years at a carbon intensity of 10 g.CO_2/kWh (chosen such that embodied carbon accounts for 20% of the lifecycle emissions).

Processor	Purchase Year	Server Count	Technology Node	Embodied Carbon ($KgCO_2$)	Performance & Power				Operational Carbon ($KgCO_2$)	Carbon (g.CO_2 /Score-Yr)		
					PassMark	TDP (W)	Cores	Threads		M	O	SCI
Xeon-Silver-4216	2020	59	14	24.15	20613	100	16	32	43.80	0.234	0.425	0.659
Xeon-Silver-4116	2019	109	14	21.18	14660	85	12	24	37.23	0.289	0.508	0.797
Xeon-E5-2640v4	2016	54	14	19.08	12472	90	10	20	39.42	0.194	0.632	0.826
Xeon-E5-2640v3	2015	65	22	19.36	11118	90	8	16	39.42	0.183	0.709	0.892
Xeon-E5-2650v2	2014	36	22	09.44	9866	95	8	16	41.61	0.096	0.844	0.939
Xeon-E5-2620-v4	2017	30	14	13.47	9193	85	8	16	37.23	0.209	0.810	1.019
Xeon-Gold-6326	2021	68	10	101.0	35270	185	16	32	81.03	0.573	0.459	1.032
Xeon-E5640	2012	47	32	11.39	3782	80	4	8	35.04	0.251	1.853	2.104
Xeon-E5620	2010	52	32	12.71	3590	80	4	8	35.04	0.253	1.952	2.205
Xeon-E5-2609-v2	2014	22	22	10.49	3369	80	4	4	35.04	0.312	2.080	2.392
Xeon-X5647	2012	82	32	13.45	4441	130	4	8	56.94	0.253	2.564	2.818
Xeon-E5520	2010	25	45	12.12	2524	80	4	8	35.04	0.343	2.777	3.120
Xeon-E5410	2008	43	65	11.75	2007	80	4	4	35.04	0.365	3.492	3.857
Xeon-E5335	2007	28	65	13.45	1549	80	4	4	35.04	0.542	4.524	5.066
Xeon-E5310	2007	20	65	14.19	1306	80	4	4	35.04	0.639	5.366	6.005
Total	–	**740**	–	**17632.71**	**8261198**	**74045**	**6204**	**11956**	–	–	–	–

4 AN ACADEMIC DATACENTER STUDY

In the previous section, we used a simple example of two servers to illustrate how different metrics, server specifications, datacenter characteristics, and accounting practices influence the *sunk carbon fallacy*. In this case study, we analyze a real-world MIT academic datacenter that supports scientific computing workloads [33, 34]. This study shows that the *sunk carbon fallacy* is not limited to simple examples but also occurs in real-world datacenters with diverse servers. Our analysis assumes that carbon-aware scheduling minimizes the total cluster-level carbon footprint—embodied and operational—when running jobs on available servers.

1 – Case Study Setup. We follow the setup from Section 3.2, with some modifications. Table 4 details the servers' specifications, which include 15 different processor types across 740 servers, with an average server age of 9.5 years. The oldest servers (`E5310`, `E5335`) are 17 years old, while the newest (`Gold-6326`) are 3 years old. Only 31.9% of servers are less than five years old. All processors are from Intel, using 64nm to 10nm technology nodes. The processors' embodied carbon ranges from 9.44 KgCO2 to 101.0 KgCO2, with a total of 17,633 $KgCO_2$. PassMark scores (multi-threaded) vary from 1306 for the oldest (`E5310`) to 35,270 for the newest (`Gold-6326`), and TDP values range from 80W (`E5310`) to 185W (`Gold-6326`).

We assume a server lifespan of 5 years. However, academic clusters often keep servers operational beyond this due to factors beyond performance and cost. We use two approaches to account for embodied carbon: 1) setting the embodied carbon of servers older than five years to 0, and 2) amortizing embodied carbon over the server's lifespan. We use the second approach in Table 4, as setting it to 0 for older servers would artificially inflate the *sunk carbon fallacy*. We use a dataset of 14 million jobs collected in 2016, from MGHPCC cluster [4, 5], including information on job submission times, end times, requested core, and memory. For comparable performance across the heterogeneous machines, we normalize the machines by thread count and create three virtual core categories shown in Figure 6: **VC1** includes 13.1% of threads with a performance score of 250–500, **VC2** includes 68.9% of threads with a score of 550–700 (2×), and **VC3** includes 18.2% of threads with a score of 750–1000 (3×). Since the largest server in our case study datacenter has 32 threads, we filter out all jobs requiring more than 32 cores.

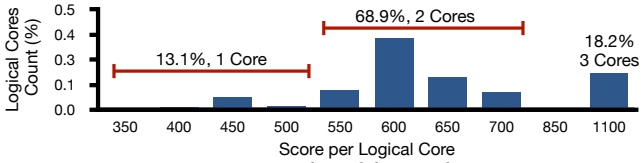

Figure 6: Normalized logical cores.

2 – Case Study Findings. 2 – Case Study Findings. Table 4 presents the `SCI` values for servers sorted by ascending `SCI`, reflecting their energy-efficient ordering. For instance, according to the `SCI` metric, `Xeon-E5-2620-v4` would be selected over `Xeon-Gold-6326`, even though the former has a 1.37× higher carbon footprint. While `Xeon-Gold-6326` is the second most energy-efficient server, it ranks 7th in `SCI`. Similar inefficiencies occur, such as choosing `Xeon-E5-2620-v4` over `Xeon-E5-2650-v2` due to lower embodied carbon. If the embodied carbon of servers older than five years is set to 0, the rankings shift even more, with the three most efficient servers—`Silver-4216`, `Gold-6326`, and `Silver-4116`—ranked 2nd, 7th, and 4th, respectively.

Though seemingly minor, these ranking changes can significantly increase the datacenter's carbon footprint when using `SCI`. We calculate the added carbon under `SCI` and `oSCI` to evaluate the cluster-level impact. Jobs are placed on

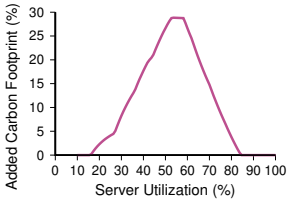

Figure 7: Embodied amortized across the lifespan.

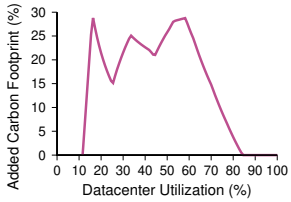

Figure 8: Embodied amortized in the first 5 years.

servers based on their submission time in a one-time placement, mimicking long-running jobs that never finish. Each job requires a specific number of virtual cores, and multiple jobs can share a server to avoid stranding resources. A trace replay and placement simulation is beyond the scope.

Figure 7 and Figure 8 show the added carbon due to SCI for two embodied carbon amortization approaches. In both cases, using SCI increases the datacenter's carbon footprint by nearly 30%, driven by the use of energy-inefficient servers. Under the first amortization approach, added carbon exceeds 5% when datacenter utilization ranges from 27% to 78%, a typical range for most datacenters. The second approach leads to even higher added carbon (often above 10%) over a wider utilization range of 13% to 80%. These results highlight how small changes in server selection order can significantly impact the overall carbon footprint. This analysis also reveals how SCI is susceptible to arbitrary choices in setting server lifespan expectations. Given that the cluster utilization in our job trace ranges from 40% to 80%, using SCI would result in a carbon footprint increase of at least 15%. Notably, the first amortization approach results in double-counting embodied carbon, which is already accounted for in the initial 5 years.

5 IMPLICATIONS AND CONCLUSION

Next, we discuss the implications of using the three carbon-based metrics in on-premise and cloud datacenters.

SCI quantifies the total carbon footprint of a functional unit by incorporating both operational and embodied emissions. While SCI is intuitive and comprehensive, it is unsuitable for all decisions. The metric assumes that any increase in a server's embodied carbon must be offset by an equal or greater reduction in operational emissions for the server to be favored over a reference. However, because embodied and operational carbon occur on different timescales, arbitrary settings for server lifespan and embodied carbon accounting can distort this ratio. As shown in Figs. 4–8, varying approaches to embodied carbon accounting and expected lifespans can non-intuitively change operational carbon.

One key aspect of SCI is that it incentivizes using older hardware, which often has a much lower embodied carbon per performance score due to being built with older, less energy-intensive technology. While advancements in smaller technology nodes have increased performance per unit area, they haven't always improved energy efficiency enough to offset the higher embodied carbon of newer servers. As shown in Table 4, this can make older servers attractive, especially once their embodied carbon has been amortized over their expected lifespan. In the worst case, this leads to older, less efficient servers being used for base demand, while newer, more efficient servers are reserved for infrequent peaks. Although SCI encourages using older servers, it inadvertently promotes a strategy where new servers are purchased but not fully utilized until they age. While extending hardware life is important, relying on older servers for base demand is not. Older hardware should be kept, but it should only be used during peak demand. Using SCI to justify increased operational carbon is counterproductive. Our analysis suggests that job scheduling should be decoupled from procurement decisions. SCI can be useful for procurement teams when replacing existing servers, helping them select new servers with lower SCI values. However, purchasing for new capabilities—such as supporting emerging workloads that require new hardware—should be SCI-agnostic. Once new servers are procured, their embodied carbon has occurred, and the focus should shift to operational carbon.

The unified approach of **tSCI** simplifies carbon cost allocation by aligning accounting and scheduling practices. However, due to variability in manufacturing processes, supply chains, and data quality, the uncertainty surrounding embodied carbon estimates makes it difficult to rely on such metrics for scheduling decisions. Introducing this uncertainty into an otherwise precise operational carbon calculation can lead to suboptimal prioritization. Also, as discussed in Section 3.2, tracking and computing tSCI over time in large-scale infrastructures, like public clouds, introduces significant overhead, limiting its practicality for real-time scheduling.

oSCI is the most effective metric for carbon-aware scheduling, as operational carbon is the primary factor that can be optimized, and hardware replacement decisions fall outside the scope of scheduling. Focusing on oSCI ensures that scheduling decisions minimize operational emissions, which is the only carbon cost that can be directly controlled after procurement. Hardware replacement, which impacts embodied carbon, should be handled separately from scheduling. By using oSCI, both on-premise and cloud datacenters can reduce operational costs by selecting the most energy-efficient servers and avoiding the *sunk carbon fallacy*.

ACKNOWLEDGEMENTS

We thank the SoCC reviewers and our shepherd, Timothy Zhu, for their valuable feedback. This work was supported in part by the NSF grants CNS-2325956, CNS-2105494, and CNS-2213636, the NSF CAREER Award CCF-2326182, the NSERC grants RGPIN-2021-03714 and DGECR-202100462, a Sloan Research Fellowship, and an Intel Research Award.

REFERENCES

[1] 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. https://doi.org/10.1145/3575693.3575754

[2] Bilge Acun, Matthew Murphy, Xiaodong Wang, Jade Nie, Carole-Jean Wu, and Kim Hazelwood. 2021. Understanding Training Efficiency of Deep Learning Recommendation Models at Scale. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. 802–814. https://doi.org/10.1109/HPCA51647.2021.00072

[3] Maria L. Alcaraz, Arash Noshadravan, Melissa Zgola, Randolph E. Kirchain, and Elsa A. Olivetti. 2018. Streamlined Life Cycle Assessment: A Case Study on Tablets and Integrated Circuits. *Journal of Cleaner Production* 200 (2018), 819–826. https://doi.org/10.1016/j.jclepro.2018.07.273

[4] Pradeep Ambati, Noman Bashir, David Irwin, and Prashant Shenoy. 2020. Waiting game: optimally provisioning fixed resources for cloud-enabled schedulers. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (Atlanta, Georgia) *(SC '20)*. IEEE Press, Article 67, 14 pages.

[5] Pradeep Ambati, Noman Bashir, David Irwin, and Prashant Shenoy. 2021. Good Things Come to Those Who Wait: Optimizing Job Waiting in the Cloud. In *Proceedings of the ACM Symposium on Cloud Computing* (Seattle, WA, USA) *(SoCC '21)*. Association for Computing Machinery, New York, NY, USA, 229–242. https://doi.org/10.1145/3472883.3487007

[6] Luiz André Barroso and Urs Hölzle. 2007. The Case for Energy-Proportional Computing. *Computer* 40, 12 (2007), 33–37.

[7] Noman Bashir, Nan Deng, Krzysztof Rzadca, David Irwin, Sree Kodak, and Rohit Jnagal. 2021. Take it to the Limit: Peak Prediction-driven Resource Overcommitment in Datacenters. In *Proceedings of the Sixteenth European Conference on Computer Systems* (Online Event, United Kingdom) *(EuroSys '21)*. Association for Computing Machinery, New York, NY, USA, 556–573. https://doi.org/10.1145/3447786.3456259

[8] Noman Bashir, Priya Donti, James Cuff, Sydney Sroka, Marija Ilic, Vivienne Sze, Christina Delimitrou, and Elsa Olivetti. 2024. The Climate and Sustainability Implications of Generative AI. *An MIT Exploration of Generative AI* (March 27 2024). https://mit-genai.pubpub.org/pub/8ulgrckc.

[9] Noman Bashir, Tian Guo, Mohammad Hajiesmaili, David Irwin, Prashant Shenoy, Ramesh Sitaraman, Abel Souza, and Adam Wierman. 2021. Enabling Sustainable Clouds: The Case for Virtualizing the Energy System. In *Proceedings of the ACM Symposium on Cloud Computing (SoCC '21)*. Association for Computing Machinery, New York, NY, USA, 350–358. https://doi.org/10.1145/3472883.3487009

[10] Noman Bashir, David Irwin, and Prashant Shenoy. 2023. On the Promise and Pitfalls of Optimizing Embodied Carbon. In *Proceedings of the 2nd Workshop on Sustainable Computer Systems* (Boston, MA, USA) *(HotCarbon '23)*. Association for Computing Machinery, New York, NY, USA, Article 15, 6 pages. https://doi.org/10.1145/3604930.3605710

[11] Noman Bashir, David Irwin, Prashant Shenoy, and Abel Souza. 2023. Sustainable computing-without the hot air. *ACM SIGENERGY Energy Informatics Review* 3, 3 (2023), 47–52.

[12] Anvita Bhagavathula, Leo Han, and Udit Gupta. 2024. Understanding the Implications of Uncertainty in Embodied Carbon Models for Sustainable Computing. In *HotCarbon Workshop on Sustainable Computer Systems*.

[13] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 1877–1901. https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[14] Andrew Chien. 2023. Embodied Carbon (EC) is a Poor Architectural Metric, Round 2. https://www.sigarch.org/why-embodied-carbon-is-a-poor-architecture-design-metric-and-operational-carbon-remains-an-important-problem/.

[15] Andrew Chien. 2023. Why Embodied Carbon is a Poor Architecture Design Metric, and Operational Carbon Remains An Important Problem. https://www.sigarch.org/why-embodied-carbon-is-a-poor-architecture-design-metric-and-operational-carbon-remains-an-important-problem/.

[16] CPU-World. 2024. CPU-World: Microprocessors / CPUs. https://www.cpu-world.com/CPUs/CPU.html. Accessed October 2024.

[17] Robin Cubitt, Maria Ruiz-Martos, and Chris Starmer. 2012. Are bygones bygones? *Theory and decision* 73 (2012), 185–202.

[18] Peter J. Denning and Ted G. Lewis. 2017. Exponential Laws of Computing Growth. *Commun. ACM* 60, 1 (January 2017), 54–65.

[19] Daniel S. Berger et al. 2023. Reducing Embodied Carbon is Important. https://www.sigarch.org/reducing-embodied-carbon-is-important/.

[20] Green Software Foundation. 2021. Software Carbon Intensity (SCI) Specification. https://sci.greensoftware.foundation/. Accessed May 2024.

[21] Anshul Gandhi, Dongyoon Lee, Zhenhua Liu, Shuai Mu, Erez Zadok, Kanad Ghose, Kartik Gopalan, Yu David Liu, Syed Rafiul Hussain, and Patrick Mcdaniel. 2023. Metrics for Sustainability in Data Centers. *ACM SIGENERGY Energy Informatics Review* 3, 3 (2023), 40–46.

[22] Wulfram Gerstner, Werner M. Kistler, Richard Naud, and Liam Paninski. 2014. *Neuronal Dynamics: From Single Neurons to Networks and Models of Cognition*. Cambridge University Press. Section 4.6: Dimensionality Reduction and Phase Plane Analysis.

[23] Viktor Gsteiger, Pin Hong Daniel Long, Yiran Jerry Sun, Parshan Javanrood, and Mohammad Shahrad. 2024. Caribou: Fine-Grained Geospatial Shifting of Serverless Applications for Sustainability. In *The 30th ACM Symposium on Operating Systems Principles (SOSP'24)*. ACM.

[24] Udit Gupta, Young Geun Kim, Sylvia Lee, Jordan Tse, Hsien-Hsin S Lee, Gu-Yeon Wei, David Brooks, and Carole-Jean Wu. 2021. Chasing Carbon: The Elusive Environmental Footprint of Computing. In *2021 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*. IEEE, 854–867.

[25] Walid A. Hanafy, Qianlin Liang, Noman Bashir, David Irwin, and Prashant Shenoy. 2023. CarbonScaler: Leveraging Cloud Workload Elasticity for Optimizing Carbon-Efficiency. *Proc. ACM Meas. Anal. Comput. Syst.* 7, 3, Article 57 (dec 2023), 28 pages. https://doi.org/10.1145/3626788

[26] HEPiX. 2024. HS23 Scores. https://w3.hepix.org/benchmarking/scores_HS23.html. Accessed October 2024.

[27] Jonathan Koomey and Samuel Naffziger. 2015. Moore's Law Might be Slowing Down, but not Energy Efficiency. *IEEE spectrum* 52, 4 (2015), 35.

[28] Baolin Li, Siddharth Samsi, Vijay Gadepally, and Devesh Tiwari. 2023. Clover: Toward Sustainable AI with Carbon-Aware Machine Learning Inference Service. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (<conf-loc>,

<city>Denver</city>, <state>CO</state>, <country>USA</country>, </conf-loc>) *(SC '23)*. Association for Computing Machinery, New York, NY, USA, Article 20, 15 pages. https://doi.org/10.1145/3581784.3607034

[29] David Lo, Liqun Cheng, Rama Govindaraju, Luiz André Barroso, and Christos Kozyrakis. 2014. Towards Energy Proportionality for Large-scale Latency-critical Workloads. *ACM SIGARCH Computer Architecture News* 42, 3 (2014), 301–312.

[30] Diptyaroop Maji, Noman Bashir, David Irwin, Prashant Shenoy, and Ramesh K Sitaraman. 2024. The Green Mirage: Impact of Location-and Market-based Carbon Intensity Estimation on Carbon Optimization Efficacy. *arXiv preprint arXiv:2402.03550* (2024).

[31] Electricity Maps. 2022. Electricity Map. https://app.electricitymaps.com/map.

[32] Peter Mattson, Vijay Janapa Reddi, Christine Cheng, Cody Coleman, Greg Diamos, David Kanter, Paulius Micikevicius, David Patterson, Guenther Schmuelling, Hanlin Tang, Gu-Yeon Wei, and Carole-Jean Wu. 2020. MLPerf: An Industry Standard Benchmark Suite for Machine Learning Performance. *IEEE Micro* 40, 2 (2020), 8–16. https://doi.org/10.1109/MM.2020.2974843

[33] MIT. 2024. Bates Research and Engineering Center. https://bateslab.mit.edu/about/. Accessed October 2024.

[34] MIT. 2024. The Massachusetts Green High Performance Computing Center (MGHPCC). https://www.mghpcc.org/. Accessed October 2024.

[35] Mixeur. 2024. X86 CPU's Guide. https://www.x86-guide.net/en/cpu.html. Accessed October 2024.

[36] Ministry of Environment Taiwan. 2023. Taiwan's Emissions Increase Slightly in 2021, Well Below 2007 Peak. https://www.moenv.gov.tw/en/8A98EC390B973CB0/c55972e9-52dc-45f1-ad97-0619c5ccd2c7. Accessed October 2024.

[37] Elsa Olivetti and Randolph Kirchain. 2012. A Product Attribute to Impact Algorithm to Streamline IT Carbon Footprinting. In *International Symposium on Environmentally Conscious Design and Inverse Manufacturing*. Springer, 747–749.

[38] OpenBenchmarking Organization. 2024. OpenBenchmarking.org - Cross-Platform, Open-Source Automated Benchmarking Platform. https://openbenchmarking.org/. Accessed May 2024.

[39] Georgios Paschos. 2019. Time-scale Separation Principle. https://paschos.net/articles/time-scale-separation-principle/. Accessed May 2024.

[40] PassMark. 2024. CPU Benchmarks. https://www.cpubenchmark.net/. Accessed October 2024.

[41] Ana Radovanovic, Ross Koningstein, Ian Schneider, Bokan Chen, Alexandre Duarte, Binz Roy, Diyue Xiao, Maya Haridasan, Patrick Hung, Nick Care, Saurav Talukdar, Eric Mullen, Kendal Smith, Mariellen Cottman, and Walfredo Cirne. 2022. Carbon-Aware Computing for Datacenters. *IEEE Transactions on Power Systems* (2022), 1–1. https://doi.org/10.1109/TPWRS.2022.3173250

[42] Prashant Shenoy. 2024. Optimizing Embodied Emissions. *ACM SIGENERGY Energy Informatics Review* 4, 1 (2024), 1–2.

[43] Abel Souza, Noman Bashir, Jorge Murillo, Walid Hanafy, Qianlin Liang, David Irwin, and Prashant Shenoy. 2023. Ecovisor: A Virtual Energy System for Carbon-Efficient Applications. In *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*. ACM, 252–265.

[44] SPEC. 2024. CPU2017 Floating Point Speed. https://spec.cs.miami.edu/cpu2017/results/cfp2017.html. Accessed October 2024.

[45] GitHub Staff. 2022. GitHub Copilot. https://github.com/features/copilot. Accessed January 2024.

[46] Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and Policy Considerations for Deep Learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 3645–3650. https://doi.org/10.18653/v1/P19-1355

[47] Thanathorn Sukprasert, Abel Souza, Noman Bashir, David Irwin, and Prashant Shenoy. 2023. Quantifying the Benefits of Carbon-Aware Temporal and Spatial Workload Shifting in the Cloud. arXiv:2306.06502 [cs.DC]

[48] Zemin Sun, Geng Sun, Long He, Fang Mei, Shuang Liang, and Yanheng Liu. 2024. A Two Time-Scale Joint Optimization Approach for UAV-assisted MEC. arXiv:2404.04597 [eess.SY]

[49] Jennifer Switzer, Gabriel Marcano, Ryan Kastner, and Pat Pannuto. 2023. Junkyard Computing: Repurposing Discarded Smartphones to Minimize Carbon. In *Proceedings of the ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS 2023)*. ACM, 400–412. https://doi.org/10.1145/3575693.3575710

[50] Sean J. Taylor. 2020. Designing and Evaluating Metrics. https://medium.com/@seanjtaylor/designing-and-evaluating-metrics-5902ad6873bf.

[51] TechPowerUp. 2024. CPU Specs Database. https://www.techpowerup.com/cpu-specs/. Accessed October 2024.

[52] John Thiede, Noman Bashir, David Irwin, and Prashant Shenoy. 2023. Carbon Containers: A System-level Facility for Managing Application-level Carbon Emissions. In *Proceedings of the 2023 ACM Symposium on Cloud Computing*. 17–31.

[53] Muhammad Tirmazi, Adam Barker, Nan Deng, Md E. Haque, Zhijing Gene Qin, Steven Hand, Mor Harchol-Balter, and John Wilkes. 2020. Borg: the Next Generation. In *Proceedings of the Fifteenth European Conference on Computer Systems* (Heraklion, Greece) *(EuroSys '20)*. Association for Computing Machinery, New York, NY, USA, Article 30, 14 pages. https://doi.org/10.1145/3342195.3387517

[54] Amin Vahdat. 2024. Societal infrastructure in the age of Artificial General Intelligence. https://www.asplos-conference.org/asplos2024/main-program/.

[55] Jaylen Wang, Daniel S Berger, Fiodar Kazhamiaka, Celine Irvene, Chaojie Zhang, Esha Choukse, Kali Frost, Rodrigo Fonseca, Brijesh Warrier, Chetan Bansal, Jonathan Stern, Ricardo Bianchini, and Akshitha Sriraman. 2024. Designing Cloud Servers for Lower Carbon. In *Proceedings of the 51st Annual International Symposium on Computer Architecture (ISCA)*.

[56] Xiaorong Wang, Clara Na, Emma Strubell, Sorelle Friedler, and Sasha Luccioni. 2023. Energy and Carbon Considerations of Fine-Tuning BERT. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

[57] Xinliang Wei, A B M Mohaimenur Rahman, Dazhao Cheng, and Yu Wang. 2023. Joint Optimization Across Timescales: Resource Placement and Task Dispatching in Edge Clouds. *IEEE Transactions on Cloud Computing* 11, 1 (2023), 730–744. https://doi.org/10.1109/TCC.2021.3113605

[58] WikiChip. 2024. WikiChip: Chips & Semi. https://en.wikichip.org/wiki/WikiChip. Accessed October 2024.

[59] World Resource Institute 1998. Greenhouse Gas Protocol. https://ghgprotocol.org/. Accessed May 2024.

[60] Carole-Jean Wu, Ramya Raghavendra, Udit Gupta, Bilge Acun, Newsha Ardalani, Kiwan Maeng, Gloria Chang, Fiona Aga, Jinshi Huang, Charles Bai, Michael Gschwind, Anurag Gupta, Myle Ott, Anastasia Melnikov, Salvatore Candido, David Brooks, Geeta Chauhan, Benjamin Lee, Hsien-Hsin Lee, Bugra Akyildiz, Maximilian Balandat, Joe Spisak, Ravi Jain, Mike Rabbat, and Kim Hazelwood. 2022. Sustainable AI: Environmental Implications, Challenges and Opportunities. In *Proceedings of Machine Learning and Systems*, D. Marculescu, Y. Chi, and C. Wu (Eds.), Vol. 4. 795–813. https://proceedings.mlsys.org/paper_files/paper/2022/file/462211f67c7d858f663355eff93b745e-Paper.pdf