Is Chiplet the Key to Greener AI Accelerators? A Quantitative Benchmarking of Real Chiplet Architectures

YUHAN SUN, KU Leuven, Belgium
JIACONG SUN, KU Leuven, Belgium
XIAOLING YI, KU Leuven, Belgium
MARIAN VERHELST, KU Leuven, Belgium

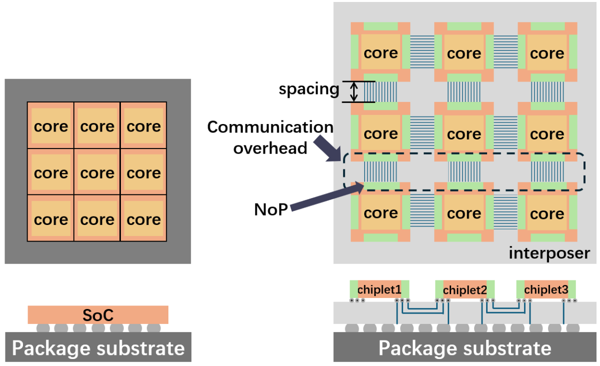
The growing carbon footprint of AI accelerators highlights the urgent need for greener hardware design strategies. Recent works point out that the chiplet-based architecture can be a more sustainable alternative to the monolithic System-on-Chip (SoC) solution due to its modular design methodology and lower design cost. However, the carbon benefit of chiplet-based accelerators has never been benchmarked quantitatively based on real hardware architectures, limiting the applicability of these works. To address the gap, we develop an analytical carbon model and simulator for the cutting-edge chiplet-based AI accelerators and conduct a thorough quantitative comparison between the chiplet and SoC solutions. The results reveal two key insights. Firstly, the chiplet solution is not universally more carbon-efficient and greener than SoCs. Though with the advantages of low design cost, an additional non-negligible carbon footprint is required due to extra interconnect area and interposer spacing, which are overlooked in existing works. Secondly, through a design space exploration across different system area and computation capacity, we reveal that chiplet-based architectures offer superior sustainability only when the functional area is relatively large (e.g., larger than 230mm^2) and the chiplet count remains moderate (typically between 4 and 9). As the number of chiplets increases further, the benefits are outweighed by packaging and interconnect overhead. In contrast, monolithic SoC designs become more favorable when the overall functional area and computation capacity are small (e.g., smaller than 141 mm^2).

CCS Concepts: • **Hardware → Integrated circuits**; • **Computing methodologies → Modeling and simulation**.

Additional Key Words and Phrases: Chiplet, System-on-Chip, Carbon Model, Carbon-Aware Scheduling, Design Space Exploration

1 INTRODUCTION

The recent widespread adoption of artificial intelligence (AI) in everyday applications has been accompanied by a rapid escalation in both model complexity and size [1–3]. To support such data- and computation-intensive workloads, super-scale AI accelerators based on system-on-chip (SoC) architectures [4–7] have become increasingly prevalent. Huge amounts of cores are typically deployed in SoC-based accelerators, as shown in Figure 1 (a), enabling high throughput and computation parallelism. While these huge monolithic accelerators have delivered impressive performance gains, they have also brought significant environmental effects. Notably, the lifecycle carbon emissions associated with running large language model inference tasks, such as BLOOM [8], on the Nvidia A100 40GB GPUs have been reported to surpass 50 tons of CO_2 equivalent emissions [9], which is roughly equal to the total emissions generated by two medium-sized passenger cars over their entire lifespans [10]. This growing environmental impact underscores the urgent need for more sustainable, energy-efficient alternatives to current monolithic SoC-based AI accelerator architectures.



(a) SoC-based accelerator

(b) Chiplet-based accelerator

Fig. 1. SoC-based (a) and chiplet-based (b) accelerator design comparison.

Recent carbon modeling studies [11, 12] pointed out chiplet-based architecture as a promising hardware candidate for greener AI accelerators. As shown in Figure 1 (b), compared to SoC-based accelerators, chiplets partition the system into smaller functional units (named chiplets), rather than integrating all components on a single die [11]. To jointly construct a multi-core architecture, chiplets are commonly interconnected through on-package links, namely Network-on-Package (NoP) [13], through silicon interposers or organic substrates. Different chiplets will be precisely placed and planted on the same substrate with uniform spacing for NoP in the middle. Chiplet architectures can inherently offer carbon cost advantages. On one hand, chiplet-based accelerators offer package-level modularity, reducing design effort and design carbon footprint by confining development to individual dies. On the other hand, fabricating smaller dies significantly improves manufacturing yields, saving material waste and decreasing fabrication carbon cost compared to SoC implementations [14].

However, prior works [11, 12] solely rely on idealized and often unrealistic assumptions, which may lead to conclusions that diverge from actual hardware behavior. For instance, with neglecting the NoP area required by chiplet communications, ECO-CHIP [11] argues that chiplets can reduce embodied carbon emissions by up to 30% compared to traditional monolithic SoCs. Yet, in the latest chiplet-based accelerators [15, 16], interconnection takes a non-negligible part of the total area, which implies a significant embodied carbon footprint. Furthermore, ECO-CHIP assumes that

---

Authors' Contact Information: Yuhan Sun, yuhan.sun@student.kuleuven.be, KU Leuven, Leuven, Belgium; Jiacong Sun, jiacong.sun@kuleuven.be, KU Leuven, Leuven, Belgium; Xiaoling Yi, xiaoling.yi@kuleuven.be, KU Leuven, Leuven, Belgium; Marian Verhelst, marian.verhelst@kuleuven.be, KU Leuven, Leuven, Belgium.

a chiplet-based hardware is partitioned into digital logics, analog logics, and memories, which are fabricated under different technology nodes. However, in current accelerator designs, the accelerator system is divided by cores, which contain all three parts and are integrated on chiplets with the same process node [17–20]. Similarly, CORDOBA [12] assumes ideal packaging conditions and neglects the interposer-related overhead. However, current technology imposes spacing constraints between chiplets, resulting in additional area overhead compared with SoCs. This packaging overhead, as shown in this paper later, can contribute 75% of the overall carbon footprint, which can cancel out the benefits of chiplets. Consequently, it remains unclear how far the conclusion in existing works deviates from real hardware results and whether chiplet-based accelerators always provide system-level benefits over monolithic SoC designs in terms of carbon efficiency.

Motivated by these gaps, this work proposes a more detailed carbon cost model that considers all the practical factors in real chiplet-based AI accelerators and systematically compares the carbon efficiency of chiplet and SoC architectures in AI accelerator design based on results collected from real chiplet processors. Experiments show that the conclusions in existing works substantially deviate from real cases, and the chiplets are not universally greener than SoC solutions. In summary, our contributions are listed as follows:

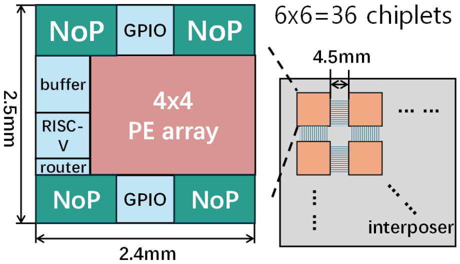
- We develop a more detailed analytical model to calculate the carbon efficiency in the total Carbon-Delay Product (tCDP) metric for real multi-core chiplet-based accelerators. (Section 3)
- By developing a simulator for SIMBA and SambaNova chiplet architectures, we experimentally show when the chiplet architecture is better or worse than SoC alternatives and how the system area and computation capacity reshape the optimal configurations. (Section 4)

2 BACKGROUND AND MOTIVATION

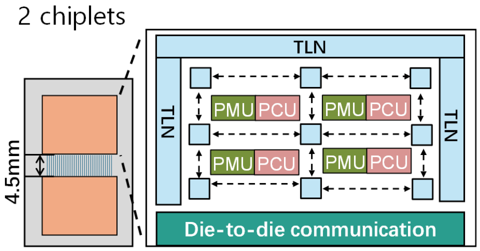
2.1 Real Chiplet-Based AI Accelerators

SIMBA [15] and SambaNova [16] stand out as two prominent chiplet-based AI accelerators developed in recent years and are used as our referenced architectures in this paper. Figure 2 illustrates their overall architecture with the technology and packaging details summarized in Table 1. All the parameters are derived from real chips' measurement results [15, 16].

As shown in Figure 2 (a), SIMBA is fabricated in 16nm CMOS technology and comprises 36 identical chiplets, each with compute and memory units. The 36 chiplets are arranged in a 6×6 mesh via a customized on-package network. Each chiplet contains 16 processing elements (PEs) for leveraging arithmetic parallelism, 4 unidirectional NoP modules for inter-die communication, and a router responsible for coordinating data exchange. In contrast, SambaNova is implemented in 5nm CMOS technology with only two functional chiplets integrated using 2.5D Chip-on-Wafer-on-Substrate (CoWoS) packaging [21], as illustrated in Figure 2 (b). The two chiplets are linked together via a specially designed die-to-die NoP communication module. Inside each chiplet, the on-chip Top Level Network (TLN) connects the chiplet tile to the host, memory, and peer-to-peer interfaces.



(a) SIMBA



(b) SambaNova

Fig. 2. Hardware architecture of the two chiplet-based accelerators. (a) SIMBA has 36 chiplets connected together in one package. (b) SambaNova has 2 chiplets in one package. This work includes all functional components in the carbon model except the blue-highlighted infrastructure (e.g., GPIO, TLN, routers).

Table 1. Architectural and packaging parameters for SIMBA [15] and SambaNova [16]

Parameter	SIMBA	SambaNova
Process node (nm)	16	5
Chiplet count	36	2
Chiplet area (mm^2)	6	650
PEs per chiplet	1024	83200
Memory per chiplet (MB)	1.38	260
NoP area per chiplet (mm^2)	2.08	84
Inter-chiplet spacing (mm)	4.5	4.5
Packaging type	Passive interposer	CoWoS

2.2 Carbon-Efficiency Evaluation Metric

To evaluate the carbon efficiency of hardware architectures, we adopt the total Carbon-Delay Product (*tCDP*) from the CORDOBA framework [12]. Extending the traditional Energy-Delay Product (EDP), *tCDP* integrates both operational and embodied carbon emissions with system latency. Unlike minimizing total carbon alone [22], which may favor low-performance designs, *tCDP* captures the trade-off between carbon impact and performance, enabling fair comparisons under carbon constraints.

Table 2. Symbol abbreviation used in proposed carbon model

Symbol	Description
$C_{em}^{(i)}$	Embodied carbon footprint of the i-th chiplet
C_{op}	Operational carbon footprint
M	Thread scheduling matrix
D_i	Execution time of the i-th pipelined threads
$E^{(i)}$	Consumed Energy of the i-th thread
CI	Operational carbon intensity

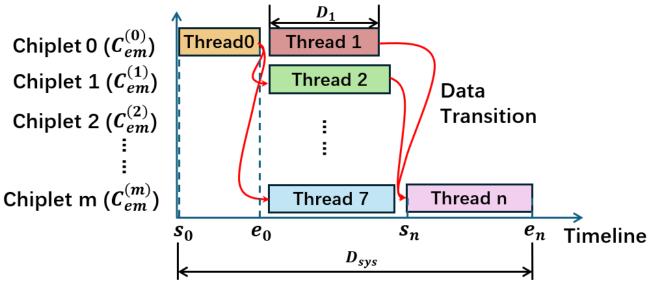


Fig. 3. Task mapping and execution timeline across chiplets. Each thread has a start time s and an end time e.

As shown in Eq. 1, *tCDP* sums embodied carbon C_{em} and operational carbon C_{op}, and multiplies the total system latency D_{sys}. Operational carbon can be expressed as the product of runtime energy consumption E and the carbon intensity CI of the process. The embodied carbon is discounted based on the D_{sys} and lifetime. For chiplet-based accelerators, C_{em} includes emissions from chiplet manufacturing, interposers, and packaging, while C_{op} accounts for compute, memory, and interconnect energy consumption.

$$tCDP = (C_{\text{em}} + C_{\text{op}}) \cdot D_{sys} = (C_{em} + E \cdot CI) \cdot D_{sys} \tag{1}$$

3 METHODOLOGY

This section presents the proposed modeling methodology used to evaluate the carbon efficiency of chiplet-based AI accelerator architectures. We will first demonstrate the workload mapping on the chiplet-based architecture and then introduce each component of our proposed detailed *tCDP* model. Table 2 summarizes the symbol abbreviations used in this section.

3.1 Workload Mapping on Chiplet-Based AI Accelerator

As the *tCDP* of a chiplet-based accelerator is closely tied to the system latency and energy consumption of the workload (as shown in Eq. 1), we will first introduce how workloads are mapped onto such accelerators. To execute a workload, we first divide it into multiple threads, which are then scheduled both spatially and temporally across the available chiplets. As illustrated in Figure 3, each chiplet is assigned one or more threads, depending on resource availability and workload characteristics. Due to data dependencies between different threads, intermediate results may need to be exchanged across chiplets through interconnect modules, introducing additional communication overhead and energy costs.

Each individual thread has a local start and end time (s and e) determined by its placement, data dependencies, and execution latency. The total system latency (D_{sys}) is defined as the elapsed time between the earliest thread start and the latest thread completion across all chiplets. Energy estimation accounts for power consumption from computation, memory access, and inter-chiplet data transmission. The system-level metrics of total execution latency and overall energy consumption collectively reflect the efficiency of a hardware architecture, and they serve as pivotal inputs for our *tCDP* evaluation.

3.2 Detailed *tCDP* Model for Chiplet-Based AI Accelerator

We will first introduce the one time *embodied carbon cost.* The total embodied carbon, C_{em}, can be distributed across the chiplets, with each chiplet taking a portion of the total as its own C_{em}. For a system containing m chiplets, the total C_{em} can be expressed as:

$$C_{em} = \sum_{i=0}^{m} C_{em}^{(i)}, \tag{2}$$

where $C_{em}^{(i)}$ represents the embodied carbon generated by the i-th chiplet.

We then derive the workload execution-related *operational carbon cost.* As shown in Figure 3, when the workload is divided into a total of n threads, each thread has its own energy consumption and execution time. As defined in Eq. 3, $\mathbf{E}$ is a vector representing the energy consumption of each thread, and $\mathbf{D}$ is a vector representing the execution time of each thread. For a single thread, s_i denotes its start time, and e_i denotes its end time. To capture the workload mapping (stated in 3.1) across chiplets, we define a scheduling matrix M of size $m \times n$, where $M[i, j] = 1$ indicates that the j-th thread is scheduled on the i-th chiplet, and $M[i, j] = 0$ otherwise.

$$\mathbf{C_{em}} = \begin{bmatrix} C_{em}^{(0)} \\ C_{em}^{(1)} \\ \cdots \\ C_{em}^{(m)} \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} E^{(0)} \\ E^{(1)} \\ \cdots \\ E^{(n)} \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} D_0 \\ D_1 \\ \cdots \\ D_n \end{bmatrix} = \begin{bmatrix} e_0 - s_0 \\ e_1 - s_1 \\ \cdots \\ e_n - s_n \end{bmatrix} \tag{3}$$

For each thread, the *tCDP* can be written as:

$$tCDP = \left(\mathbf{M}^\top \cdot \mathbf{C_{em}} + CI \cdot \mathbf{E}\right)^\top \odot \mathbf{D} \tag{4}$$

As discussed in Section 3.1, the total execution time of the system, denoted by D_{sys}, is defined as the maximum interval between the earliest start time and the latest finish time among all n threads. This is formally expressed in Eq. 5.

$$D_{sys} = \max_{n} e_{(i)} - \min_{n} s_{(i)} \tag{5}$$

In summary, the system-level *tCDP* is given by Eq. 6. $tCDP_{sys}$ accounts for the total embodied carbon C_{em} of the entire m chiplets, and total operational carbon cost C_{op} based on the energy consumption of the total n threads:

$$
\begin{aligned}
tCDP_{sys} &= C_{em} \cdot D_{sys} + C_{op} \cdot D_{sys} \\
&= \left(\sum_{i=0}^{m} C_{\text{em}}^{(i)} + \sum_{i=0}^{n} E^{(i)} \cdot CI \right) \cdot D_{sys} \\
&= \left(\sum_{i=0}^{m} C_{\text{em}}^{(i)} + \sum_{i=0}^{n} E^{(i)} \cdot CI \right) \cdot \left(\max_{n} e_{(i)} - \min_{n} s_{(i)} \right) \qquad (6)
\end{aligned}
$$

4 EXPERIMENTS

This section analyzes how the most carbon-efficient chiplet organizations are affected under the SIMBA [15] and SambaNova [15] chiplet architectures. Our goal is to identify, by keeping the same system area or the computation capacity, when and what chiplet organizations are more carbon-friendly over monolithic SoC designs. In addition, beyond the SIMBA and SambaNova configurations, the final case study explores the broader hardware design space and demonstrates how the total system area influences both the optimal chiplet architecture and its carbon efficiency.

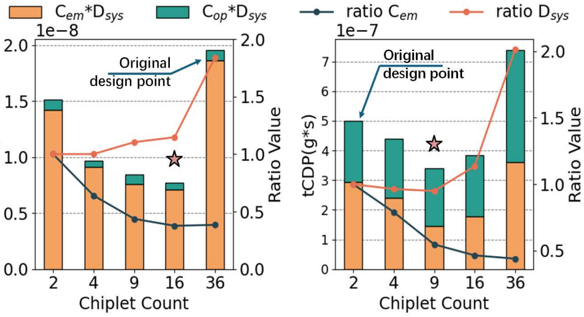
4.1 Experimental Setup

To enable the carbon footprint analysis of SIMBA and SambaNova systems, we developed a simulator based on the chiplet configurations and chip measurement results listed in Table 2. The simulator calculates the carbon efficiency by the method stated in Section 3. Unlike prior works that model chiplet systems using idealized separations (e.g., logic/SRAM/analog split), we partition the chiplets in the same pattern as the real fabricated chip [15, 16]. Each chiplet comprises multiple standard computation cores and its dedicated L1/L2 memories, with NoP area and interconnection overhead considered across different chiplets. Carbon modeling parameters are extracted from ECO-CHIP [11], including carbon intensity values for chip manufacturing, packaging, operational energy, and basic design-related overheads. The embodied carbon (C_{em}) is estimated based on the model presented in ECO-CHIP [11] over a two-year lifetime for each hardware. The Stream simulator [23] is used to determine the optimal thread scheduling across chiplets based on the genetic algorithm, and the ZigZag simulator [24, 25] is leveraged to estimate the energy and latency of every single thread.

We evaluate the system performance and carbon footprint with the ResNet50 network [1] as the targeting workload benchmark. In the following subsections, we analyze how the optimal chiplet organizations are affected by keeping a constant system area and a constant computation capacity.

4.2 Chiplets with Constant System Area

As AI accelerators are often constrained with a certain area budget, we first evaluate *how the chiplet count and hardware organizations affect the carbon efficiency under the same system area.* The bar chart in Figure 4 shows the carbon cost breakdown and how carbon efficiency scales for the SIMBA and SambaNova architecture, with the optimal chiplet count and the original referenced system configuration [15, 16] highlighted by stars and texts. As shown in the breakdown, overall *tCDP* is largely dominated by the product $C_{em} \cdot D_{sys}$. Notably, the chiplet configuration that achieves optimal carbon efficiency differs from those used in existing SIMBA and SambaNova hardware, highlighting the opportunity for improving the carbon efficiency of current chiplet systems.



(a) SIMBA

(b) SambaNova

Fig. 4. *tCDP* and breakdown for SIMBA (a) and SambaNova (b) chiplet architectures under the same system area with different chiplet counts. The blue curve and the orange curve represent the normalized C_{em} (ratio C_{em}) and the normalized D_{sys} (ratio D_{sys}), respectively.

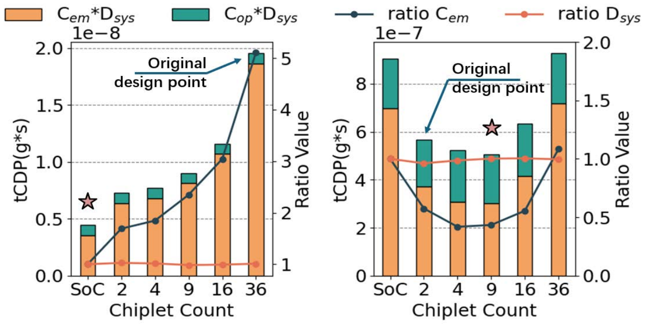
For the SIMBA architecture, as depicted in Figure 4 (a), the overall *tCDP* improves by 40% as the chiplet count increases from 2 to 16 (a 4 × 4 configuration). Beyond this point, however, *tCDP* rises sharply at 36 chiplets. This trend results from a reduction in embodied carbon due to improved chiplet yield when the count is below 16, as shown by the blue curve representing normalized embodied carbon (C_{em}). As a result, *tCDP* decreases in line with the embodied carbon footprint. However, the yield improvement gradually flattens beyond 16 chiplets. In contrast, since the NoP and interconnect overheads grow with the chiplet count, the computation capacity drops, leading to an increased system latency (D_{sys}) and *tCDP*.

Similarly, for the SambaNova architecture shown in Figure 4 (b), the overall *tCDP* gradually decreases, reaching a minimum at 9 chiplets due to the yield improvement. Beyond this point, further partitioning provides no additional yield benefit and instead reduces the computational capacity, resulting in an increase of the overall *tCDP*.

4.3 Chiplets with Constant Computation Capacity

Since industrial AI accelerators often have strict requirements for computational capacity and throughput, with more flexibility in the system area, this subsection evaluates *the structural overhead introduced by chiplet partitioning under a constant computation capacity constraint.* Under this setup, the number of processing elements remains fixed, but the total silicon and packaging area may increase with the chiplet count due to the additional interconnect and packaging overhead.

Overall, as shown in Figure 5, the system-level execution time D_{sys} remains relatively stable, which is a consequence of the preservation of compute resources. However, the embodied carbon C_{em}



(a) SIMBA

(b) SambaNova

Fig. 5. *tCDP* and breakdown for SIMBA (a) and SambaNova (b) under constant computation capacity. The blue curve and the orange curve represent the normalized C_{em} (ratio C_{em}) and the normalized D_{sys} (ratio D_{sys}), respectively.

varies significantly with chiplet fragmentation, becoming the dominant factor driving *tCDP* growth. For SIMBA, the SoC configuration demonstrates the optimal carbon efficiency in comparison with other counterparts, indicating that the SoC possesses competitiveness when the compute area is small. In contrast, SambaNova achieves its lowest *tCDP* at the 9-chiplet configuration, which offers an optimal balance between improved manufacturing yield and packaging complexity. Both systems exhibit a sharp increase in *tCDP* when the chiplet count reaches 36, primarily due to substantial interconnect and packaging overhead.

A detailed breakdown of *tCDP* components corresponding to the fabricated chiplet architectures in [15, 16] and SoC alternatives, shown in Figure 6, reveals that the divergent trends stem from differences in architectural area composition. SIMBA features a relatively small system area footprint, making it more suitable for monolithic SoC integration; increasing the chiplet count introduces additional 74.8% packaging and 7% NoP overhead that significantly raises the overall embodied carbon footprint. In contrast, SambaNova has a much larger system area, and moderate chiplet partitioning-specifically a 2 or 4-chiplet configuration-effectively improves the manufacturing yield without incurring substantial packaging overhead (only 9.5%). This observation aligns with earlier findings on the carbon and yield benefits of chiplet-based integration for large compute area systems [11].

4.4 *tCDP* Evaluation Across Varied System Area

We further evaluate *the carbon efficiency of SIMBA and SambaNova across varying chiplet counts and system area configurations*, as illustrated in Figure 7. The results indicate that, in both SIMBA and SambaNova architectures, configurations with fewer chiplets, typically around 4, generally yield superior carbon efficiency although SoC-based design is most sustainable in SIMBA when the compute area is relatively small. This trend holds even as the total compute area increases, highlighting the environmental advantages of moderate chiplet partitioning.

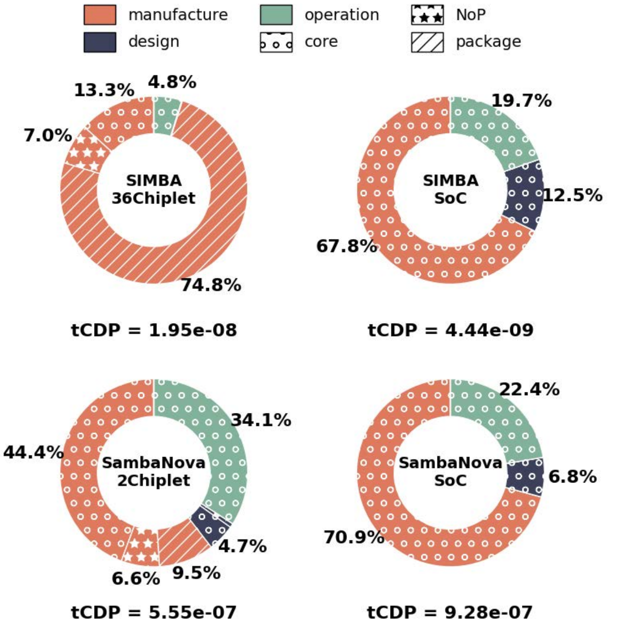


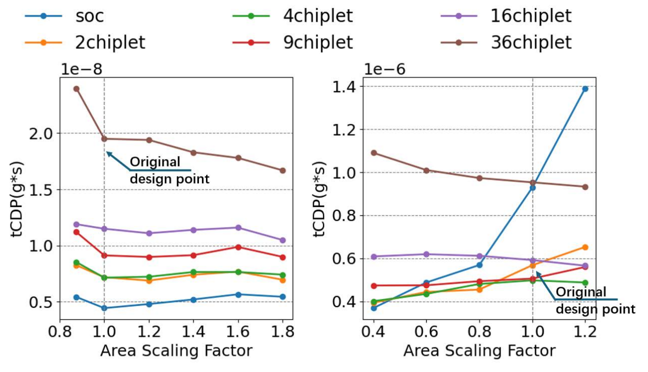
Fig. 6. *tCDP* breakdown of the SIMBA (top) and SambaNova (bottom) architectures between the chiplet implementations and the corresponding SoC alternatives.

An additional observation arises in the SIMBA 36-chiplet configuration (brown curve in Figure 7 (a)), where *tCDP* decreases noticeably with increasing system area. This is because, with the system area increasing, the increase in embodied carbon C_{em} is relatively modest, while execution time D_{sys} improves significantly, yielding better carbon efficiency. In contrast, SambaNova's SoC-based design (blue curve in Figure 7 (b)) exhibits a sharp increase in *tCDP* as the system area grows, driven by the substantial embodied carbon associated with scaling a monolithic die.

These results in general suggest that while moderate chiplet partitioning (e.g., 4-9 chiplets) achieves robust carbon efficiency across a range of area budgets, excessive fragmentation or monolithic scaling both incur environmental penalties. Notably, our exploration results reveal that the most carbon cost-efficient chiplet architecture diverges from the current chiplet design choice, delivering key insight for future chiplet design.

5 CONCLUSION

We present a detailed analytical performance and carbon model for the cutting-edge chiplet-based AI accelerators, considering all the practical factors. We conduct a thorough quantitative comparison between the chiplet and SoC solutions based on two well-known SIMBA and SambaNova chiplet architectures. The findings of this study demonstrate that moderate chiplet partitioning, typically comprising 4 to 9 chiplets, achieves optimal carbon efficiency across a range of scenarios. In the context of a constant system area, an increase in the chiplet count has been observed to result in elevated



(a) SIMBA

(b) SambaNova

Fig. 7. *tCDP* for SIMBA (a) and SambaNova (b) under different system areas (normalized) and chiplet count.

packaging and interconnect overhead as well as decreased computation ability, thereby leading to an increase in *tCDP* beyond 16 chiplets. In the context of a constant computation capacity, having more chiplets leads to a substantial increase in embodied carbon due to the expansion of the area overhead.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[2] Min Zhang and Juntao Li. A commentary of gpt-3 in mit technology review 2021. *Fundamental Research*, 1(6):831–833, 2021.

[3] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

[4] Yang Hu, Xinhan Lin, Huizheng Wang, Zhen He, Xingmao Yu, Jiahao Zhang, Qize Yang, Zheng Xu, Sihan Guan, Jiahao Fang, et al. Wafer-scale computing: Advancements, challenges, and future perspectives [feature]. *IEEE Circuits and Systems Magazine*, 24(1):52–81, 2024.

[5] Drago Ignjatović, Daniel W Bailey, and Ljubisa Bajić. The wormhole ai training processor. In *2022 IEEE International Solid-State Circuits Conference (ISSCC)*, volume 65, pages 356–358. IEEE, 2022.

[6] NVIDIA Corporation. Nvidia hopper architecture in-depth, 2022. Accessed: 2025-05-13.

[7] Apple Inc. Apple introduces m4 pro and m4 max, 2024. Accessed: 2025-05-13.

[8] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. 2023.

[9] Alexandra Sasha Luccioni, Sylvain Viguier, and Anne-Laure Ligozat. Estimating the carbon footprint of bloom, a 176b parameter language model. *Journal of Machine Learning Research*, 24(253):1–15, 2023.

[10] Ricardo plc. Lifecycle emissions from cars. Technical Report MC-P-11-15a, Low Carbon Vehicle Partnership (LowCVP), 2011. Prepared for the Low Carbon Vehicle Partnership.

[11] Chetan Choppali Sudarshan, Nikhil Matkar, Sarma Vrudhula, Sachin S Sapatnekar, and Vidya A Chhabria. Eco-chip: Estimation of carbon footprint of chiplet-based architectures for sustainable vlsi. In *2024 IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, pages 671–685. IEEE, 2024.

[12] Mariam Elgamal, Doug Carmean, Elnaz Ansari, Okay Zed, Ramesh Peri, Srilatha Manne, Udit Gupta, Gu-Yeon Wei, David Brooks, Gage Hills, et al. Cordoba: Carbon-efficient optimization framework for computing systems. In *2025 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pages 1289–1303. IEEE, 2025.

[13] Mariam Musavi, Emmanuel Irabor, Abhijit Das, Eduard Alarcon, and Sergi Abadal. Communication characterization of ai workloads for large-scale multi-chiplet accelerators. *arXiv preprint arXiv:2410.22262*, 2024.

[14] The advantage of amd's chiplet architecture. Technical report, Advanced Micro Devices, Inc., 2022. White Paper.

[15] Yakun Sophia Shao, Jason Cemons, Rangharajan Venkatesan, Brian Zimmer, Matthew Fojtik, Nan Jiang, Ben Keller, Alicia Klinefelter, Nathaniel Pinckney, Priyanka Raina, et al. Simba: scaling deep-learning inference with chiplet-based architecture. *Communications of the ACM*, 64(6):107–116, 2021.

[16] Raghu Prabhakar, Junwei Zhou, Darshan Gandhi, Youngmoon Choi, Mahmood Khayatzadeh, Kyunglok Kim, Uma Durairajan, Jeongha Park, Satyajit Sarkar, and Jinuk Luke Shin. 16.4: Sambanova sn40l: A 5nm 2.5 d dataflow accelerator with three memory tiers for trillion parameter ai. In *2025 IEEE International Solid-State Circuits Conference (ISSCC)*, volume 68, pages 288–290. IEEE, 2025.

[17] Yuan Li, Ahmed Louri, and Avinash Karanth. Scaling deep-learning inference with chiplet-based architecture and photonic interconnects. In *2021 58th ACM/IEEE Design Automation Conference (DAC)*, pages 931–936. IEEE, 2021.

[18] d-Matrix. Corsair: Scaling generative inference with digital in-memory compute. Technical report, d-Matrix Corporation, November 2024.

[19] Gianna Paulin, Paul Scheffler, Thomas Benz, Matheus Cavalcante, Tim Fischer, Manuel Eggimann, Yichao Zhang, Nils Wistoff, Luca Bertaccini, Luca Colagrande, et al. Occamy: A 432-core 28.1 dp-gflop/s/w 83% fpu utilization dual-chiplet, dual-hbm2e risc-v-based accelerator for stencil and sparse linear algebra computations with 8-to-64-bit floating-point support in 12nm finfet. In *2024 IEEE Symposium on VLSI Technology and Circuits (VLSI Technology and Circuits)*, pages 1–2. IEEE, 2024.

[20] Tenstorrent Inc. Tt-metalium programming guide. https://github.com/tenstorrent/tt-metal/blob/main/METALIUM_GUIDE.md, May 2025.

[21] John H Lau. Recent advances and trends in advanced packaging. *IEEE Transactions on Components, Packaging and Manufacturing Technology*, 12(2):228–252, 2022.

[22] Udit Gupta, Mariam Elgamal, Gage Hills, Gu-Yeon Wei, Hsien-Hsin S Lee, David Brooks, and Carole-Jean Wu. Act: Designing sustainable computer systems with an architectural carbon modeling tool. In *Proceedings of the 49th Annual International Symposium on Computer Architecture*, pages 784–799, 2022.

[23] Arne Symons, Linyan Mei, Steven Colleman, Pouya Houshmand, Sebastian Karl, and Marian Verhelst. Stream: A modeling framework for fine-grained layer fusion on multi-core dnn accelerators. In *2023 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, pages 355–357. IEEE, 2023.

[24] Linyan Mei, Pouya Houshmand, Vikram Jain, Sebastian Giraldo, and Marian Verhelst. Zigzag: Enlarging joint architecture-mapping design space exploration for dnn accelerators. *IEEE Transactions on Computers*, 70(8):1160–1174, 2021.

[25] Jiacong Sun, Pouya Houshmand, and Marian Verhelst. Analog or digital in-memory computing? benchmarking through quantitative modeling. In *2023 IEEE/ACM International Conference on Computer Aided Design (ICCAD)*, pages 1–9. IEEE, 2023.