2024 IEEE 26th Electronics Packaging Technology Conference (EPTC) | 979-8-3315-2200-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/EPTC62800.2024.10909776

Cost-Performance Co-Optimization for the Chiplet Era

Alexander Graening[*], Darayus Adil Patel[†], Giuliano Sisto[†], Erwan Lenormand[†], Manu Perumkunnil[†], Nicolas Pantano[†],
Vinay B.Y. Kumar[†], Puneet Gupta[*], Arindam Mallik[†]

[*]University of California, Los Angeles, California, USA [†]imec, Kapeldreef 75, 3001 Leuven, Belgium

agraening@ucla.edu, darayus.adil.patel@imec.be, giuliano.sisto@imec.be, erwan.lenormand@imec.be,
manu.perumkunnil@imec.be, vinay.kumar.baapanapalliyadaiah@imec.be, puneet@ee.ucla.edu, arindam.mallik@imec.be

Abstract

Chiplet technologies allow for greater flexibility in system design through a wide range of system configuration options spanning integration schemes (monolithic, 2.5D, 3D), heterogeneity in technology nodes, and partitioning of system resources. Each such configuration has implications on figures of merit such as cost, power and performance. Optimizing for one metric may come at the expense of the other two. During the initial architecture exploration and design planning stage, it is critical to conduct cost-performance co-optimization using models that cover the entire spectrum of configuration options to make an informed choice. This work presents an evaluation using a framework that models both cost and performance simultaneously for chiplet-based systems, enabling analysis of the impact of various system-wide architectural configurations. We analyze a representative scale-out system for high-performance computing workloads to investigate how factors like integration, partitioning, technology node choices, and system size affect power, performance, and cost. The results demonstrate that the optimal design choices vary depending on these factors, highlighting the insights early stage chiplet design space exploration can offer.

Introduction

Advanced integration and packaging will drive the scaling of computing systems in the next decade. Chiplet systems [1, 2] are becoming increasingly prevalent in the industry for high performance computing and are under active exploration for automotive systems [3]. NVIDIA [4], Intel [5], and AMD [6] have all released chiplet-based products. The diversity of integration choices (monolithic, 2.5D, 3D) [2] and partitioned die sizes for chiplet based systems enlarges the design space and widens the attainable cost and performance profile. This necessitates cost-performance co-optimization to determine ideal system configuration [7].

With respect to performance, chiplets allow for integration of very large systems. Instead of integrating these systems as multiple packaged chips, we can integrate them as a chiplet system with near monolithic performance. While integrating large systems in a single package tends to improve performance, disaggregating monolithic designs into multiple chiplets can hurt performance due to the increased signal lengths resulting from die separation. Generally splitting a design and integrating in a 2D or 2.5D manner will result in worse or equivalent performance, but in the case of 3D stacking there can be a performance improvement over the monolithic case due to reducing signal length by changing long cross-die connections into relatively short vertical connections.

Chiplet-based design can offer significant reduction in system cost due to improving yield, however, there are other confounding factors to consider. A chiplet design requires more engineering work upfront to appropriately partition the system into the optimal number and kinds of chiplets. Chiplet systems are more expensive to assemble and package than monolithic designs, particularly if complicated stacking or interposers are required. Additionally, splitting a design into chiplets requires adding die-to-die interface IOs that increase total area and power for the design.

Chiplets also have other benefits that are more difficult to quantify. Creating a new design based on previously designed chiplets can be faster than creating the design from scratch, reducing time to market. Thus, reuse is an important design consideration. Additionally, if it is easier to customize different product instances with minimal silicon waste, it can be practical to offer more variations of a product family by offering different combinations of chiplets with minimal re-design.

In addition to the inherent benefits and drawbacks of different chiplet architecture options, the cost and performance of chiplet systems depends on factors such as inter-chiplet IO scheme, substrate type, and bump pitch. These factors are important to consider early in the design process. This requires detailed cost and performance modeling to make informed design decisions.

In this work, we motivate the need for both cost and performance modeling during the architectural definition phase to identify cost-performance co-optimal points in the design space. For a representative scale-out system designed for HPC workloads, we study the impact of partitioning, integration choices, technology node, and system size on individual and combined metrics involving performance, power, and cost. Our results illustrate the variation in co-optimal design points, highlighting the insights that performance and cost modeling for early stage chiplet design space can offer.

Ther rest of the paper is organized as follows. First, we discuss related works, then we describe the example system we used for our study. Next, we describe our modeling framework with details about both our cost model and our performance model. After this we show the results of our system analysis on a variety of metrics.

Related Work

The manufacturing cost of dies and silicon interposers has been previously studied in [8] and [9]. However, these studies do not account for the cost for substrates, die-to-die (D2D) overhead, or non-recurring engineering (NRE) costs. Quantitative cost modeling for chiplet-based designs has been

979-8-3315-2200-1/24/$31.00 ©2024 IEEE

published in “Chiplet Actuary” [9] and “Chiplets: How Small is too Small” [10]. These analytical cost models account for multiple elements of chiplet cost including manufacturing yield, packaging cost, IO requirements, technology nodes, and many other factors. In particular, the work in [10] provides an analytical framework to determine the system size above which disaggregating an SoC into smaller chiplets yields cost benefits and below which disaggregating will tend to carry a cost penalty.

The total cost of ownership (TCO) benefit of employing chiplet-based systems in building AI supercomputers for serving large language model (LLM) workloads has been established in the work on Chiplet Cloud [11] where the authors demonstrate that chip costs dominate TCO. The study in [12] explores the question of how to partition compute resources across chiplets as well as the trade-off in performance versus cost and sustainability. The impact of inter-chiplet network in 2.5D integration on system performance for deep learning inference workloads has been studied in [13]. A design space exploration framework for chiplet based processors is shown in [14], while [15] proposes HISIM, a benchmarking tool for chiplet-based heterogeneous integration that evaluates the performance of monolithic, 2.5D and 3D systems.

It is pertinent to note that none of the existing works in literature simultaneously explore the performance and cost variation for chiplet-based systems wherein the integration scheme, technology node, system partitioning, and system size are simultaneously varied.

System Description

Fig. 1 illustrates components of a single node of a scale-out system architecture. We consider a single node of this system for cost-performance evaluation, and it is composed of:

- Compute arrays containing control cores, RISC-V based SIMT compute cores (CCs), and local SRAM.
- Data processing unit (DPU) attached to a DRAM (HBM) and a storage system.
- Network processing unit (NPU) building a distributed storage system.

Nodes are integrated into boards and the full system is composed of a network of boards. Cost and performance are intertwined and impacted by the system architecture and physical SoC configuration. We examine a few physical configuration options (Fig. 2) to evaluate the trade-offs therein. Cost is estimated for a single package and components that are shared between packages and thus packaged separately – i.e., the colored blocks in Fig. 1 – are ignored since they do not change across the chosen configurations. The rest of the logic and memory silicon (including HBM) is split up according to the configurations shown in Fig. 2.

The four chiplet-configuration options shown in Fig. 2 contain the same HBM chiplet and have different configurations for the compute cores and memory. In the first column, labeled “fewer chiplets,” everything except the HBM is integrated in a single die/stack. This consists of a single monolithic chiplet and a 3D stack of logic on memory respectively. The “more chiplets” column contains both a more aggressively split 2.5D integration option and a version including 3D stacking. In both cases, the DPU is placed as its own independent chiplet while the memory and cores are split into two groups either integrated as 2 separate chiplets or as two separate 3D stacks.

We evaluated these configurations for both 7nm and 5nm. Additionally, we considered a variety of system sizes by varying the number of compute cores per board and boards per system. The area measures we used for this study are from physical aware synthesis.

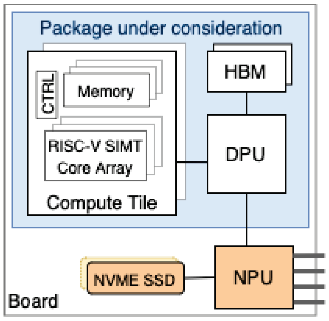

Fig. 1. System architecture under evaluation consists of compute tiles, data processing unit (DPU), HBM, network processing unit (NPU), and storage (SSD).

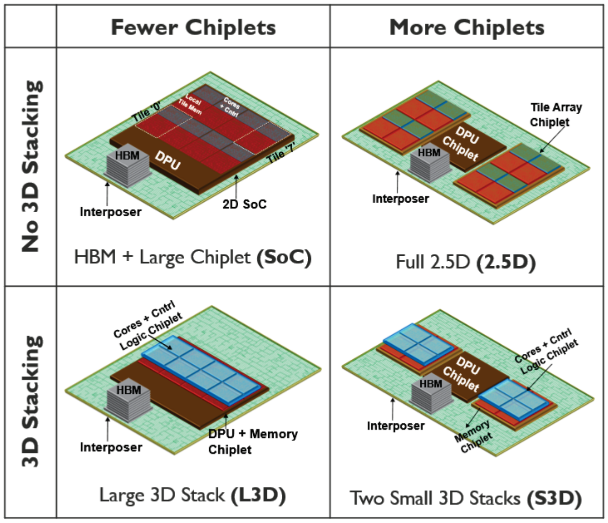

Fig. 2. Configurations for cost-performance co-optimization. 2.5D splits SoC into 3 chiplets and S3D splits L3D stack into two 3D stacks plus one additional chiplet.

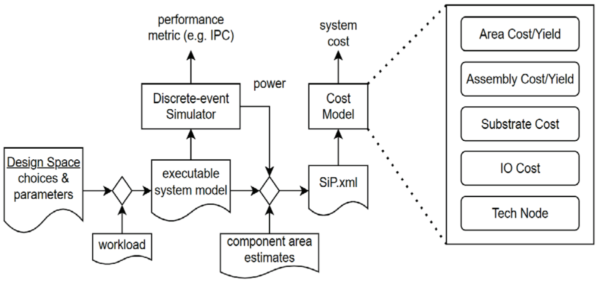

Fig. 3. Evaluation framework.

System Modeling

Fig. 3 depicts our evaluation framework that simultaneously models cost and performance for chiplet-based systems used for analyzing the impact of different system-wide architectural configurations, integration scheme, technology node, system partitioning, system size and workloads. The framework supports modeling and evaluating a configurable high-performance computing (HPC) system comprised of compute core/s, memory and network components. The inherent cost and performance models embedded in our framework are detailed below.

Cost Model

To analyze system cost, we used the open source chiplet cost model previously described in [10]. This cost model uses a nested stack of chip class objects meant to allow flexible cost analysis of arbitrary chiplet based designs. The model is fully parameterized so users can define custom technologies and processes for their studies. This model is open source at https://github.com/nanocad-lab/cost_model_chiplets.

Fig. 4 depicts the required input parameters for the cost model. Bonding pitch, assembly machine costs, assembly yields and other parameters are included in the assembly technology process. Assembly cost and yield are modeled to be dependent on the nature of bonding (e.g., solder reflow vs. die-to-wafer thermocompression bonding) and bump pitches which influence tool cost. The IO cell includes the additional area required to drive a signal over a longer distance between chiplets compared to a monolithic design as well as the additional energy per bit. IOs are only placed in regions which meet the reach requirement for the cell type. Total chiplet area is calculated as the larger of either core area plus IO cell area or the area required by bumps for signaling and power. The layer definition contains parameters such as cost per mm^2 and defect density required to compute the cost and yield of dies. Yield is computed using the negative binomial yield model [16]. The wafer process contains reticle size and dicing information to improve the accuracy of the cost per die. We assume pre-bond known-good-die testing and ignore test cost. Our cost model currently does not incorporate IP, license, and board costs.

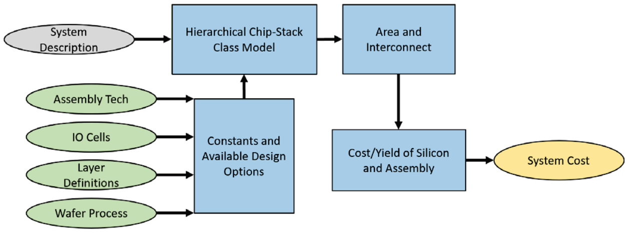
System Description
Hierarchical Chip-Stack Class Model
Area and Interconnect
Assembly Tech
IO Cells
Layer Definitions
Wafer Process
Constants and Available Design Options
Cost/Yield of Silicon and Assembly
System Cost

Fig. 4. Cost model overview. The model provides a wide range of parameters that influence cost.

The negative binomial yield model used in this work uses parameters representative of advanced CMOS nodes. Most parameters are scaled down from their correspondent value for the 10nm node. For all options, the portion of area considered critical is 60%, the clustering factor parameter in the yield model is assumed to be equal to 2 and the stitching yield as 0.5. A few of the parameters we used for yield and cost calculations that vary by process technology node are shown in Table 1. In Fig. 5, we show the portion of cost for each configuration that comes from the costs and yield impacts of the chiplet configuration. Cost of assembly and packaging configuration are important to consider for chiplet systems.

Table 1. Parameter values used to differentiate between 5nm and 7nm cost and yield for the chiplet-based cost model.

Parameter	Description	7nm	5nm
Cost per mm^2	Cost of the wafer divided by the area of the wafer (\$/mm^2).	0.1322	0.2502
Defect density	Density of defects on the full fabricated wafer (#defects/mm^2).	0.008	0.0127
Lithography percent	Costs from time spent on the lithography tool, used to scale the costs relative to number of exposures.	0.27	0.3
NRE mask cost	Non-recurring engineering cost for the needed lithography masks (\$).	1000000	3000000

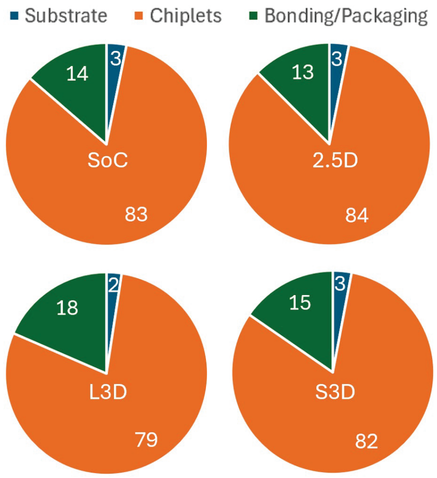

Fig. 5. Package cost breakdown (%). Bonding/packaging is for yield plus material cost not including substrate.

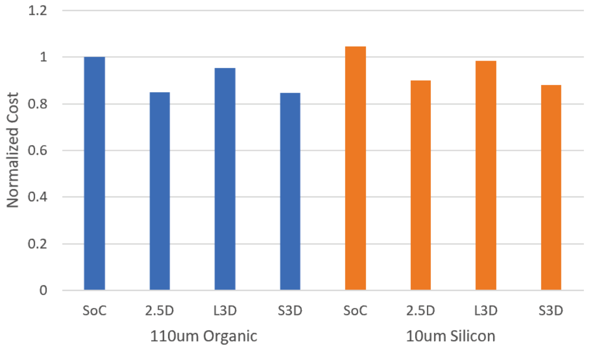

Fig. 6. Substrate cost trends for each configuration.

Organic substrates are typically cheaper than silicon, but at the expense of having much larger bump pitch. We looked at an organic substrate with chiplets bonded simultaneously using a reflow style bonding and compared this to a silicon substrate with a much lower pitch using thermal compression bonding where each die is placed and bonded individually. We compared switching from the 110um pitch organic substrate to the 10um pitch silicon substrate in Fig. 6 and Fig. 7. The silicon substrate cases are more expensive than organic. The cost of the silicon substrate is higher than the almost negligible cost of the organic substrate, but this is the primary difference between the cases. Fig. 6 shows that this case study is not pitch-bound as the cost of the organic substrate case remains the same for a smaller bump pitch and the cost of the silicon substrate case remains the same for a larger bump pitch. Silicon substrates with small bump pitch make more sense in cases that are pitch limited with a large number of IOs.

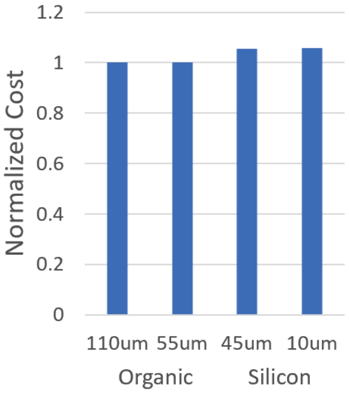

Fig. 7. Cost with bump pitches and substrates for 2.5D case.

Performance Model

A detailed, functionally correct and cycle approximate executable model of the system is used for cost-performance codesign (Fig. 3) and studied with representative workloads. The performance model has relevant parameter sensitivity to account for different chiplet configuration options (Fig. 2) i.e. latency and bandwidth between logic/memory in 2.5D vs. 3D and with D2D interfaces over UCIe. This work uses large sparse matrix-vector multiplication workloads distributed across the entire system measuring Giga floating-point operations per second (GFLOPS/s) in steady state as a performance metric. The system with multiple interconnected boards is simulated using a parallel discrete event simulator.

It may be noted that the cost and performance models are decoupled, and any other system performance model whose performance estimates can be sensitive to such configuration parameters can also be used. For instance, our model uses a 3D SRAM performance model for 3D configuration (accounting for latency and bandwidth improvements) vs. non-3D configurations.

System-level power estimation is done as a function of system parameters and the underlying components. This uses power estimates via characterization of individual function blocks within the components with EDA tools targeting relevant technology nodes. Additionally, variation in IO power as a function of system partitioning (Fig. 2) is also accounted for in system-level power. Hierarchical area estimation also follows a similar approach.

Results and Discussion

While the results and observations here correspond to a specific system architecture, our conclusions and methodology can be helpful for informing other chiplet system architecture configuration choices.

Performance measurements in Figs. 8-13 correspond to executing large sparse matrix-vector multiplication (SPMV) distributed across various parameterized configurations of the system. In these figures, a tile is a unit of design containing several cores within a package (Fig. 1). More tiles mean a larger package which impacts package cost.

Fig. 8 shows the normalized system performance for the different system configurations and package sizes. Fig. 9-13 consider 5nm and 7nm variants with an iso-performance assumption. It is also interesting to note that the performance benefit of 3D stacking in the L3D case is mostly lost in the S3D case compared to the SoC case due to the increased signal distance in inter-chiplet communication.

Fig. 9 shows normalized system power. Since we assume iso-performance for 7nm and 5nm, we do see a power benefit for 5nm. We do not observe any appreciable power difference between the chiplet configurations, but the options with more chiplets (2.5D and S3D) and so more interconnect power have slightly higher power than the options with fewer chiplets (SoC and L3D), matching our expectations. Note that power scales better for 5nm than for 7nm.

Fig. 10 shows the normalized system cost. Here we see the impact of the system size on the preferred number of chiplets. The L3D case (2 stacked chiplets + HBM) is cheapest for the smallest 2-tile system while the S3D (5 total chiplets + HBM) is better for the larger 8-tile system. We also see the SoC (1 chiplet + HBM) is already impacted by yield in the 2-tile case and is consistently the worst option for cost.

In a power-constrained application, we care about how many operations can be executed on a power budget. Fig. 11 shows the performance per Watt for our system. Note that L3D consistently outperforms SoC and S3D consistently outperforms 2.5D in the same technology due to the performance benefit of 3D stacking memory on compute. We also see all 5nm options are better than all 7nm options for the 8-tile case due to the spike in power we see in Fig. 9.

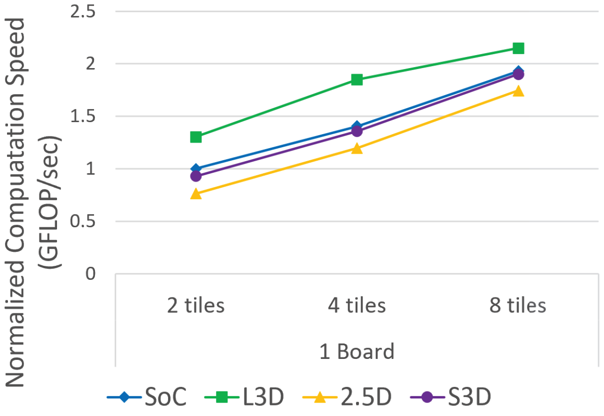

Fig. 8. Normalized system performance.

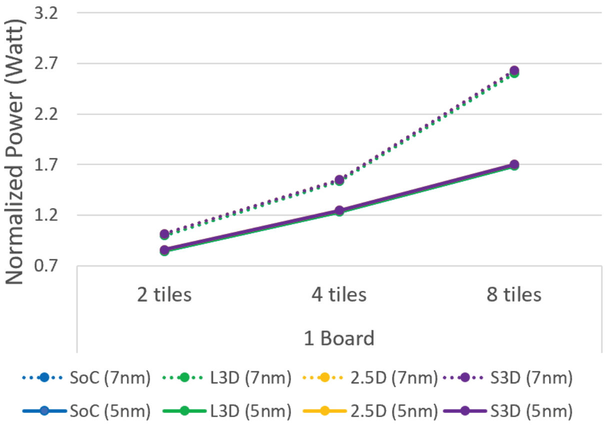

Fig. 9. Normalized system power.

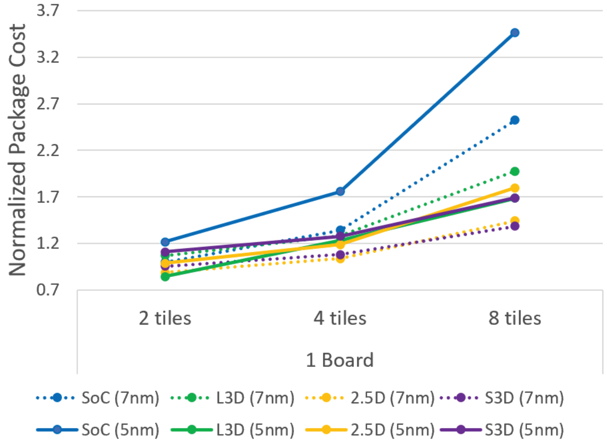

Fig. 10. Normalized system cost

If our application is cost constrained rather than power, we care about maximizing the operations we can get for a certain monetary budget. Fig. 12 shows the impact on performance per dollar. Here we see the 3D stacked cases behaving well due to their increased performance and we see a swap between the higher performing L3D case and the cheaper S3D case for the 8-tile package due to the much better yield in the S3D case. It is also worth noting that 5nm does not compare well to 7nm in this study since it is more expensive, and we assumed iso-performance.

If we want to optimize for all 3 metrics, then we can look at the performance per Watt per dollar. Fig. 13 shows that L3D is the best in terms of this metric for 2-tile and 4-tile systems, but S3D in 7nm is the best for the 8-tile system. This is largely the result of the performance per dollar we saw in Fig. 12, but it is interesting to note how the gap has closed between 5nm and 7nm with the inclusion of power. Again, we see 5nm get a noticeable edge due to the spike in power for 7nm in the 8-tile case shown in Fig. 9.

These measurements show how co-design with a target workload can lead to different conclusions dependent on size of system and choice of optimization target among cost, power, and performance.

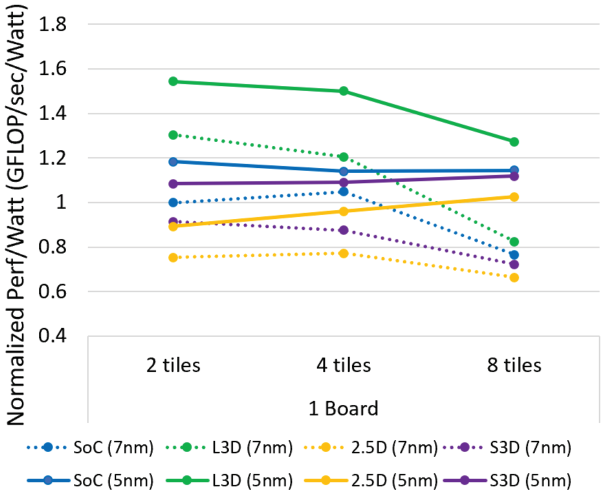

Fig. 11. Normalized performance per Watt.

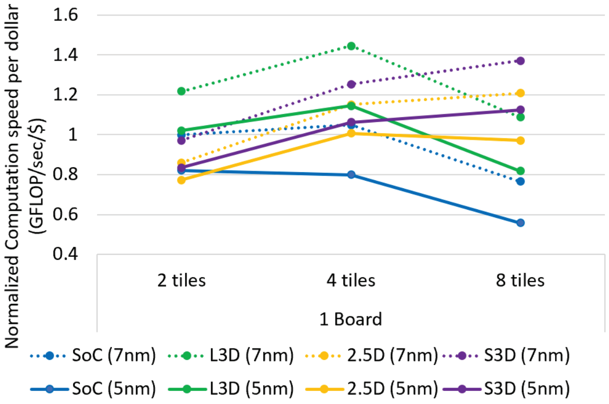

Fig. 12. Normalized performance per dollar.

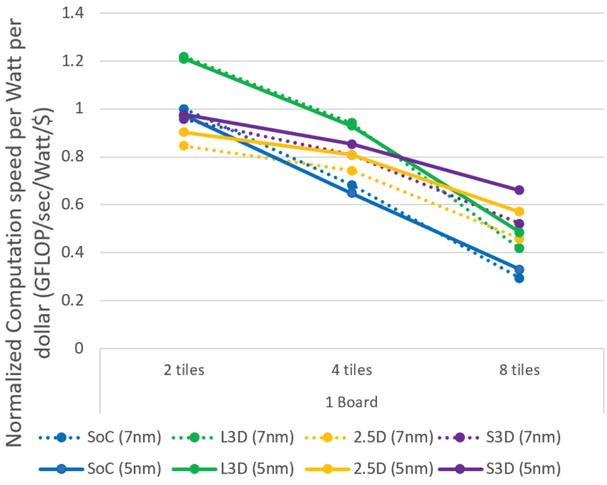

Fig. 13. Normalized performance per Watt per dollar.

Conclusions

Early cost, power, and performance co-optimization for chiplet systems is critical since changing the chiplet configuration gets more expensive later into design. Detailed system modeling can be used to allow an informed choice early in the design process. We provide insights on how doing this for a realistic system can steer 2.5D and 3D design decisions. Our cost model is open source and can be used together with any other detailed system performance model.

Our results allow us to draw several conclusions:

- Advanced packaging (silicon, small bump pitch) is not necessary in our system, but might become necessary if the inter-chiplet bandwidth or number of HBMs increases.
- Normalized package cost and chipletization benefit increase with system size due to yield.
- Small systems where chiplet yield is less of a driving factor favor fewer chiplets due to assembly cost.
- 3D stacking improves performance over monolithic, but performance equalizes if we split to multiple 3D stacks to improve cost and yield.

While our co-optimal design points are specific to system architecture, technology, and workload dynamics, the insights of our study highlight the benefits that performance and cost modeling for early stage chiplet design space exploration can offer.

Acknowledgments

This work is supported, in part, by NSF and DARPA/SRC CHIMES JUMP 2.0 Center.

References

1. Chiplets: Piecing Together the Next Generation of Chips https://www.imec-int.com/en/articles/chiplets-piecing-together-next-generation-chips-part-i
2. Li, T.; Hou, J.; Yan, J.; Liu, R.; Yang, H.; Sun, Z. Chiplet Heterogeneous Integration Technology—Status and Challenges. *Electronics* 2020, *9*, 670. https://doi.org/10.3390/electronics9040670
3. “Automotive chiplet program,” imec. Accessed: Sep. 15, 2024. [Online]. Available: https://www.imec-int.com/en/expertise/cmos-advanced-and-beyond/compute/automotive-chiplet-program
4. TechPowerUp. NVIDIA H100 SXM5 96 GB. Techpowerup.com https://www.techpowerup.com/gpu-specs/h100-sxm5-96-gb.c3974 (2024).
5. “Intel Data Center GPU Max Series Overview,” Intel. Accessed: Sep. 13, 2024. [Online]. Available: https://www.intel.com/content/www/us/en/developer/articles/technical/intel-data-center-gpu-max-series-overview.html
6. S. Naffziger *et al*., "Pioneering Chiplet Technology and Design for the AMD EPYC™ and Ryzen™ Processor Families : Industrial Product," *2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA)*, Valencia, Spain, 2021, pp. 57-70, doi: 10.1109/ISCA52012.2021.00014.
7. Pal, S., Mallik, A. & Gupta, P. System technology co-optimization for advanced integration. *Nat Rev Electr Eng* 1, 569–580 (2024). https://doi.org/10.1038/s44287-024-00078-x
8. Dylan Stow, Itir Akgun, Russell Barnes, Peng Gu, and Yuan Xie. 2016. Cost analysis and cost-driven IP reuse methodology for SoC design based on 2.5D/3D integration. In ICCAD.
9. Yinxiao Feng and Kaisheng Ma. 2022. Chiplet actuary: a quantitative cost model and multi-chiplet architecture exploration. In Proceedings of the 59th ACM/IEEE Design Automation Conference (DAC '22). Association for Computing Machinery, New York, NY, USA, 121–126. https://doi.org/10.1145/3489517.3530428
10. A. Graening, S. Pal and P. Gupta, "Chiplets: How Small is too Small?," 2023 60th ACM/IEEE Design Automation Conference (DAC), San Francisco, CA, USA, 2023, pp. 1-6, doi: 10.1109/DAC56929.2023.10247947.
11. Peng, Huwan, et al. "Chiplet cloud: Building ai supercomputers for serving large generative language models." *arXiv preprint arXiv:2307.02666* (2023).
12. Zhang, Shiqing, et al. "Balancing performance against cost and sustainability in multi-chip-module GPUs." *IEEE Computer Architecture Letters* (2023).
13. Sharma, Harsh, et al. "Achieving datacenter-scale performance through chiplet-based manycore architectures." *2023 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2023.
14. S. Pal, D. Petrisko, R. Kumar and P. Gupta, "Design Space Exploration for Chiplet-Assembly-Based Processors," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 28, no. 4, pp. 1062-1073, April 2020, doi: 10.1109/TVLSI.2020.2968904.
15. Z. Wang *et al*., "Benchmarking Heterogeneous Integration with 2.5D/3D Interconnect Modeling," *2023 IEEE 15th International Conference on ASIC (ASICON)*, Nanjing, China, 2023, pp. 1-4, doi: 10.1109/ASICON58565.2023.10396377.
16. W. Kuo and T. Kim, "An overview of manufacturing yield and reliability modeling for semiconductor products," *Proceedings of the IEEE*, vol. 87, no. 8, pp. 1329-1344, Aug. 1999, doi: 10.1109/5.775417.