Design Space Exploration for Optimal Silicon-Efficient Server Chiplets

Ayan Chakraborty[†], Ali Ansari[†], Yuanlong Li[†], Shanqing Lin[†], Arunkrishna AMS[†], Alireza Foroodnia[†]
August Ning[†], Mohammad Alian[‡], Michael Ferdman[§], Pejman Lotfi-Kamran[†¶], Babak Falsafi[†]
EcoCloud, EPFL[†], Cornell University[‡], Stony Brook University[§], IPM[¶]

Abstract

Datacenter growth is driving unprecedented demand for both electricity and physical infrastructure. Today's servers typically rely on wide out-of-order (OoO) cores running at high nominal frequencies to meet SLOs, but rising power, silicon, memory, and cooling constraints call for rethinking this design point. We present a methodology for designing power-density-optimized CPU chiplets for server workloads. Our results show that optimal operating frequency can be up to 50% lower for air-cooled systems, and chiplets may favor in-order (InO) cores even for SLO-sensitive services, contrary to conventional trends. Overall, power-density-optimized chiplets improve performance by 1.7×–6.1× over conventional out-of-order baseline chiplets.

1 Introduction

Datacenter growth has been exponential in recent years thanks to the explosion of global cloud services, the unprecedented computing demands of AI, and the slowdown in Moore's Law. Even in the AI era, CPUs remain crucial; datacenter demand is projected to grow by 1.7× for conventional (CPU-based) workloads and by 3.5× for AI (CPU- and accelerator-based) workloads by 2030 [18].

Contemporary server CPUs employ high-frequency (≥3 GHz), 4–6-wide out-of-order cores. These CPUs typically have large on-chip caches which act as dim silicon to meet air-cooling power density limits [9, 12, 27]. As servers shift from air to liquid cooling, chips can operate at higher power densities, and industry trends are moving toward designing CPUs with higher core counts, higher single-thread performance, and even higher operating frequency [10].

However, datacenter operators face pressing operational and capital constraints that are unlikely to ease soon. A May 2025 report from the International Energy Agency projects global datacenter electricity use to grow by 16% annually, exceeding 950 TWh by 2030 [15]. Meanwhile, rising server costs and the slowdown of Moore's Law have pushed architects toward larger CPUs to improve performance per socket. Yet manufacturing cost grows superlinearly with die area, making continued area scaling increasingly unsustainable.

The 3rd Workshop on Hot Topics in System Infrastructure (HotInfra'26), co-located with ISCA'26, June 28, 2026, Raleigh, NC, USA

These trends in turn influence the optimal architecture and operation of future server CPUs. Given current power limitations, it is essential to maximize performance per power. Designs must also optimize performance per area for high silicon and memory utilization. Computer architects must design for both of these (often competing) metrics while meeting SLO and power density constraints.

We show that when optimizing performance for a given power density (unifying area and power constraints), design space parameters interact in ways that challenge prior assumptions and motivate new design choices. To the best of our knowledge, this is the first work to design power-density-optimized chiplets for server workloads by comprehensively exploring the design space across all major circuit, architectural, and system-level factors that influence the trade-off between performance, power, and area.

We combine simulation with analytic models to identify core, frequency, and transistor type configurations on the Pareto frontier between performance per area and performance per power using CloudSuite [5], DCPerf [26], and DeathStarBench [11]. Using these Pareto-optimal configurations, we search the chiplet design space including on-chip SRAM and queuing semantics to maximize SLO-constrained performance under a target power density. Finally, we evaluate the optimal chiplets using full-system cycle-accurate simulation and queuing models to demonstrate higher performance over baseline air- and liquid-cooled chiplets.

This paper makes the following contributions:

- We show air-cooled chiplets built with high-threshold logic transistors running at 1.5-2.0 GHz outperform 3.0 GHz baselines by 1.7-6.1× while meeting SLO.
- We show single-queue semantics enables chiplets with in-order cores to become competitive even for workloads with tight SLOs.
- We show liquid-cooled chiplets still see benefits from reducing frequency. Chiplets operating at 2.8 GHz outperform 3.4 GHz baselines by 1.7-5.8×.

2 Optimal Chiplet Design Space Parameters

Prior work [2, 21, 25] has studied how core complexity and frequency affect IPC, area, and power. Here, we focus on less explored chiplet design-space parameters and the interactions that arise especially when considering SLO constraints.

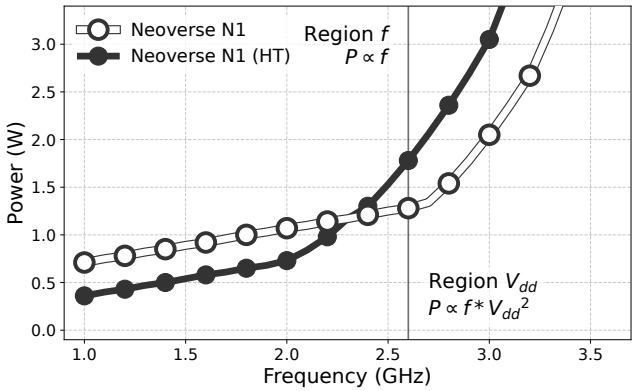

Figure 1. Power against frequency modeling for an off-the-shelf Neoverse N1 and an all-HT Neoverse N1 core.

2.1 Frequency & Transistor Type

Transistors typically operate in the *superthreshold* region, where V_{dd} is well above threshold voltage [4]. We focus on this region because near-threshold operation faces performance, robustness, and yield challenges [8, 24]. Superthreshold operation further has two sub-regions. In *region* V_{dd}, voltage and frequency scale together until reaching V_{min}, reducing dynamic power roughly cubically with frequency. Below V_{min}, in *region f*, voltage is fixed and dynamic power scales only linearly with frequency. Figure 1 illustrates this behavior for a modeled 7nm ARM Neoverse N1 core.

At low frequencies, leakage becomes 30-50% of total power, largely due to low-threshold-voltage (LT) transistors in latency-sensitive logic and upper-level SRAMs. We observe that a hypothetical all-high-threshold-voltage (HT) core can eliminate leakage and result in lower total power in region f as shown in Figure 1. Although HT transistors are unsuitable for high frequencies because they require a higher V_{dd} (thereby increasing dynamic power), they are an attractive choice for low-frequency designs. Thus, transistor type and frequency must be considered jointly.

2.2 Impact of SLO

A tight SLO favors high single-thread performance, but improving single-thread performance increases core power and area superlinearly. Therefore, the most area- and power-efficient core depends on SLO strictness. Figure 2 illustrates this interaction for a Web Serving workload [5], showing the Pareto frontier between performance per area and performance per power. The plotted frontier does not account for L2 and LLC SRAM area and power.

Figure 2 (left) shows the Pareto frontiers for the target 300 ms SLO, while Figure 2 (right) shows the effect of relaxing the SLO by 4×. Under a tight SLO, high single-thread performance is required, so OoO cores dominate InO cores in both performance per area and performance per power. As the SLO relaxes, cores can run at lower frequencies and InO cores become Pareto-optimal instead. The dashed lines

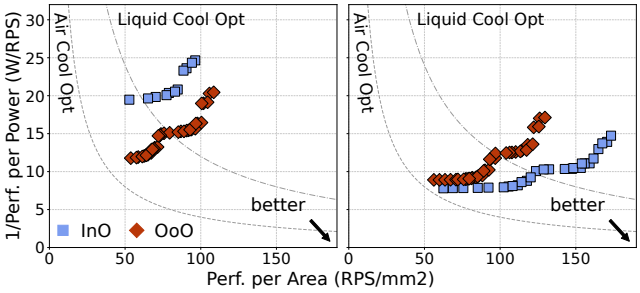

Figure 2. Pareto-optimal cores running Web Serving under the target SLO (left) and a four-times relaxed SLO (right).

Table 1. Single-core design space parameters

Parameter	Range
Core Microarchitecture	InO, OoO
Pipeline Width	1, 2, 3, 4
L1 (4-way) size	8 KB, 16 KB, 32 KB, 64 KB
BTB (4-way) entries	256, 512, 1K, 2K, 4K, 8K, 16K
SMS (16-way) entries	None, 1K, 2K, 4K, 8K, 16K
Clock Frequency	1 GHz - 3 GHz
Transistor Type	LT, HT

indicate air- and liquid-cooling power-density targets. All cores exceed the air-cooling target, indicating that chiplet designs need dim SRAM to reduce overall power density.

Most prior work [2, 13, 21] studies core-complexity tradeoffs without SLOs. While related work [7] uses Pollack's Rule and queuing models to study SLO-aware core choices, core performance and area do not always scale quadratically [9], and complexity may not reflect performance for memory-bound workloads. These factors motivate revisiting core-complexity tradeoffs for SLO-constrained workloads.

2.3 Choice of Queuing Semantics

A chiplet with n cores can use either multi-queue or single-queue semantics. In the multi-queue model (used by contemporary servers), requests are distributed across n independent per-core queues, so performance scales as $n\times$ single-core performance. In the single-queue model [6, 14, 16, 17, 19, 22], all requests enter a shared logical queue served by all cores. Queuing delay decreases sharply as core count increases, thereby favoring high core-count chiplets. To our knowledge, this is the first work to evaluate how queuing semantics affect the optimal chiplet design point.

3 Evaluation

3.1 Workloads & Methodology

We evaluate four representative workloads: Graph Analytics (GA), a batch workload without SLO; Data Caching (DC) and Web Search (WSEA), monolithic workloads with a relaxed 1 ms SLO and a tight 200 ms SLO respectively; and Media Microservices (MMS), a microservice-based workload with a tight 100 ms SLO. For non-SLO workloads, performance is measured by throughput. For workloads with

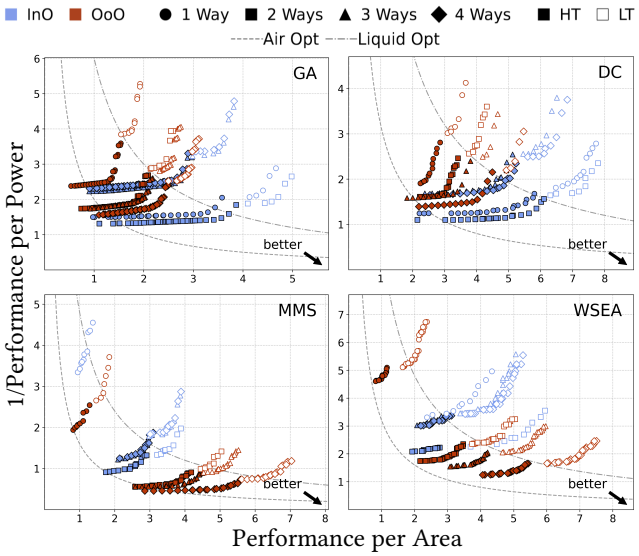

Figure 3. Single-core Pareto-optimal frontiers. Axes are linearly scaled per workload to aid comparison.

SLOs, performance is measured by the maximum throughput achieved while meeting the specified SLO. Other workloads from CloudSuite and DCPerf exhibit similar trends.

We create a single-core CPI model derived from Hardavellas et al. [12] to explore the design space (see Table 1). We use functional and cycle-accurate timing simulation [20] to extract model parameters. The CPI model, together with frequency and instruction count measurements, is used to estimate average service time across all design-space configurations. For workloads with SLOs, we combine the estimated service time with queuing models to compute performance. We then sweep the design space to identify optimal chiplet configurations for air and liquid cooling. Finally, we simulate these configurations using the sampling methodology [28] and use the resulting IPC measurements with our queuing models to obtain final performance estimates.

3.2 Pareto-optimal Core Design

Figure 3 shows single-core Pareto frontiers for the workloads. We observe that 2-way InO cores are Pareto-optimal for non-SLO workloads (such as GA), corroborating prior work [2]. Analytics workloads are memory-bound; the limited MLP gains from OoO cores are outweighed by their area and power overheads. InO cores continue to dominate OoO cores for lax SLO workloads (such as DC, 24× zero-load tail latency). DC has low ILP and MLP, making InO cores inherently more silicon-efficient for this workload, and its SLO target is insufficiently stringent to make OoO cores competitive. For MMS and WSEA (both with 6× zero-load tail latency), OoO cores dominate InO cores. WSEA is compute-bound and, when combined with a strict SLO target, strongly favors high single-thread performance. For MMS, the already strict SLO target decomposes into stricter per-service latency targets, thus favoring high single-thread performance.

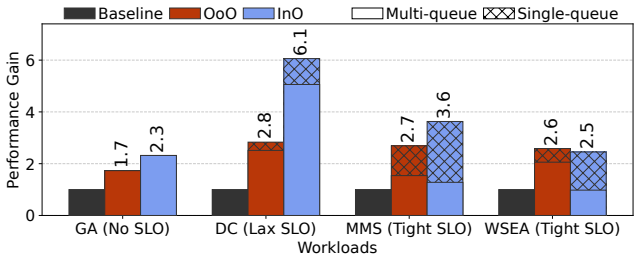

Figure 4. Perf. gain of air-cooled OoO and InO optimized chiplets. Air-cooled optimized chiplets use HT transistors.

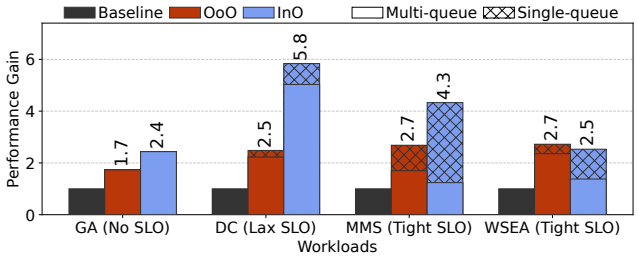

Figure 5. Perf. gain of liquid-cooled OoO and InO optimized chiplets. Liquid-cooled optimized chiplets use LT transistors.

3.3 Pareto-optimal Chiplet Design

We select the Pareto-optimal core configurations identified in our single-core design-space exploration, reserve space for a reasonably sized LLC (1 MB per four cores from working set analysis) and sweep frequency, core count, transistor type, and queuing semantics to identify optimal chiplet designs respecting area, power, and memory bandwidth budgets (taken from a scaled-down Zen 3 chiplet [3]). Air-cooled baseline uses 4-way OoO cores at 3 GHz [1], while the liquid-cooled baseline uses the same cores at 3.4 GHz [23]. Figure 4 and Figure 5 show the performance gain of our optimal chiplets over the baselines for air- and liquid-cooling respectively.

3.3.1 Air-cooled Chiplets. For non-SLO workloads (GA), the superior silicon efficiency of InO cores directly translates into higher overall performance for the optimal InO chiplet relative to the optimal OoO chiplet. Because these workloads do not have SLOs, their performance is independent of whether a multi-queue or single-queue system is used. As single-core performance is not a concern, the optimal frequency reduces to 1.5 GHz (with HT transistors) for both the InO and OoO chiplets. At this point, all cores on each chiplet are powered on and both designs are area-bound.

For SLO-constrained workloads, performance and optimal frequency depend on the queuing system. In a multi-queue system, n cores behave as n independent queues, so single-core silicon efficiency directly translates to chiplet performance, as observed across workloads. The InO chiplet performs better for lax SLO workloads, while the OoO chiplet performs better for tight SLO workloads. For both chiplets, lax SLO workloads are optimal at 1.5 GHz with all cores active, whereas tight SLO workloads are optimal at 2 GHz with

25% of cores powered off to meet single-thread performance needs. HT transistors are optimal in all cases.

Moving from a multi-queue to a single-queue system drastically changes performance trends: both chiplets are optimal at 1.5 GHz (using HT transistors) with all cores active, and the InO chiplet outperforms the OoO chiplet for almost all workloads while remaining competitive for WSEA. A single queue provides a global view of all cores and can assign each request to any available core, improving load balancing and minimizing queuing delays. The benefit is larger for the InO chiplet because it has 3× more cores, increasing the likelihood that an incoming request finds an idle core.

3.3.2 Liquid-cooled Chiplets. The higher power budget makes the chiplets area-bound rather than power-bound, allowing higher frequencies. The optimal frequency increases to 2.8 GHz (using LT transistors) with all cores active but still 17% below the 3.4 GHz baseline. Key trends in core complexity and queuing semantics remain unchanged.

4 Conclusion

We show that server chiplets can operate at substantially lower frequencies while still meeting SLOs. The saved power can instead be used to power more cores, reclaiming dark silicon in power-constrained CPUs. While OoO cores remain important for tight SLO workloads under conventional multi-queue systems, single-queue semantics shifts the optimum toward many smaller InO cores.

Acknowledgments

This work was supported in part by the Swiss National Science Foundation (SNSF) under Project No. 200021/212757, "Unified Accelerators for Post-Moore Machine Learning". We also acknowledge support from Intel through the projects "CHIMP: HW/SW Co-design Techniques for Multi-objective Optimization of Heterogeneous 2.5D/3D Chiplets" and "Virtual Memory for Post-Moore Servers". Additionally, this work benefited from support from "UrbanTwin: An Urban Digital Twin for Climate Action—Assessing Policies and Solutions for Energy, Water and Infrastructure". The authors thank the funding agencies and industrial partners for their support.

References

[1] Jean-Luc Aufranc. 2020. *Ampere Altra Announces 80-Core Arm Neoverse N1 Server Processor & Reference Designs.* https://www.cnx-software.com/2020/03/04/ampere-altra-announces-80-core-arm-neoverse-n1-server-processor-reference-designs/ CNX Software (cnxsoft).

[2] Omid Azizi, Aqeel Mahesri, Benjamin C. Lee, Sanjay J. Patel, and Mark Horowitz. 2010. Energy-performance tradeoffs in processor architecture and circuit design: a marginal cost analysis. In *Proceedings of the 37th Annual International Symposium on Computer Architecture* (Saint-Malo, France) *(ISCA '10)*. Association for Computing Machinery, New York, NY, USA, 26–36. https://doi.org/10.1145/1815961.1815967

[3] Thomas Burd, Wilson Li, James Pistole, Srividhya Venkataraman, Michael McCabe, Timothy Johnson, James Vinh, Thomas Yiu, Mark Wasio, Hon-Hin Wong, Daryl Lieu, Jonathan White, Benjamin Munger, Joshua Lindner, Javin Olson, Steven Bakke, Jeshuah Sniderman, Carson Henrion, Russell Schreiber, Eric Busta, Brett Johnson, Tim Jackson, Aron Miller, Ryan Miller, Matthew Pickett, Aaron Horiuchi, Josef Dvorak, Sabeesh Balagangadharan, Sajeesh Ammikkallingal, and Pankaj Kumar. 2022. Zen3: The AMD 2nd-Generation 7nm x86-64 Microprocessor Core. In *2022 IEEE International Solid-State Circuits Conference (ISSCC)*, Vol. 65. 1–3. https://doi.org/10.1109/ISSCC42614.2022.9731678

[4] Anantha P. Chandrakasan, William J. Bowhill, and Frank Fox (Eds.). 2001. *Design of High-Performance Microprocessor Circuits.* IEEE Press, New York.

[5] CloudSuite 4.0 [n. d.]. CloudSuite 4.0. https://www.cloudsuite.ch/. Accessed: 2025-11-11.

[6] Alexandros Daglis, Mark Sutherland, and Babak Falsafi. 2019. RPC-Valet: NI-Driven Tail-Aware Balancing of µs-Scale RPCs. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems* (Providence, RI, USA) *(ASPLOS '19)*. Association for Computing Machinery, New York, NY, USA, 35–48. https://doi.org/10.1145/3297858.3304070

[7] Christina Delimitrou and Christos Kozyrakis. 2018. Amdahl's law for tail latency. *Commun. ACM* 61, 8 (July 2018), 65–72. https://doi.org/10.1145/3232559

[8] Ronald G. Dreslinski, Michael Wieckowski, David Blaauw, Dennis Sylvester, and Trevor Mudge. 2010. Near-Threshold Computing: Reclaiming Moore's Law Through Energy Efficient Integrated Circuits. *Proc. IEEE* 98, 2 (2010), 253–266. https://doi.org/10.1109/JPROC.2009.2034764

[9] Hadi Esmaeilzadeh, Emily Blem, Renee St. Amant, Karthikeyan Sankaralingam, and Doug Burger. 2011. Dark silicon and the end of multicore scaling. In *Proceedings of the 38th annual international symposium on Computer architecture*. 365–376.

[10] Luigi Fusco, Mikhail Khalilov, Marcin Chrapek, Giridhar Chukkapalli, Thomas Schulthess, and Torsten Hoefler. 2024. Understanding data movement in tightly coupled heterogeneous systems: A case study with the Grace Hopper superchip. *arXiv preprint arXiv:2408.11556* (2024).

[11] Yu Gan, Yanqi Zhang, Dailun Cheng, Ankitha Shetty, Priyal Rathi, Nayan Katarki, Ariana Bruno, Justin Hu, Brian Ritchken, Brendon Jackson, Kelvin Hu, Meghna Pancholi, Yuan He, Brett Clancy, Chris Colen, Fukang Wen, Catherine Leung, Siyuan Wang, Leon Zaruvinsky, Mateo Espinosa, Rick Lin, Zhongling Liu, Jake Padilla, and Christina Delimitrou. 2019. An Open-Source Benchmark Suite for Microservices and Their Hardware-Software Implications for Cloud & Edge Systems. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems* (Providence, RI, USA) *(ASPLOS '19)*. Association for Computing Machinery, New York, NY, USA, 3–18. https://doi.org/10.1145/3297858.3304013

[12] Nikos Hardavellas, Michael Ferdman, Babak Falsafi, and Anastasia Ailamaki. 2011. Toward dark silicon in servers. *IEEE Micro* 31, 4 (2011), 6–15.

[13] Urs Hölzle. 2010. Brawny Cores Still Beat Wimpy Cores, Most of the Time. *IEEE Micro* (2010). https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/36448.pdf

[14] Stephen Ibanez, Alex Mallery, Serhat Arslan, Theo Jepsen, Muhammad Shahbaz, Changhoon Kim, and Nick McKeown. 2021. The nanoPU: A Nanosecond Network Stack for Datacenters. In *15th USENIX Symposium on Operating Systems Design and Implementation (OSDI 21)*. USENIX Association, 239–256. https://www.usenix.org/conference/osdi21/presentation/ibanez

[15] International Energy Agency. 2025. *Energy and AI.* Technical Report WEO-Special Report. International Energy Agency, Paris, France. https://iea.blob.core.windows.net/assets/601eaec9-ba91-4623-819b-4ded331ec9e8/EnergyandAI.pdf Lead authors: Thomas Spencer, Siddharth Singh..

[16] Rishabh Iyer, Musa Unal, Marios Kogias, and George Candea. 2023. Achieving Microsecond-Scale Tail Latency Efficiently with Approximate Optimal Scheduling. In *Proceedings of the 29th Symposium on Operating Systems Principles* (Koblenz, Germany) *(SOSP '23)*. Association for Computing Machinery, New York, NY, USA, 466–481. https://doi.org/10.1145/3600006.3613136

[17] Kostis Kaffes, Timothy Chong, Jack Tigar Humphries, Adam Belay, David Mazières, and Christos Kozyrakis. 2019. Shinjuku: Preemptive Scheduling for {μsecond-scale} Tail Latency. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*. 345–360.

[18] Erikhans Kok, Johan Rauer, Pankaj Sachdeva, Piotr Pikul, Dave Sutton, and Rawad Hasrouni. 2025. Scaling bigger, faster, cheaper data centers with smarter designs. *McKinsey & Company* (2025). https://www.mckinsey.com/industries/private-capital/our-insights/scaling-bigger-faster-cheaper-data-centers-with-smarter-designs Accessed: April 5, 2026.

[19] Nikita Lazarev, Shaojie Xiang, Neil Adit, Zhiru Zhang, and Christina Delimitrou. 2021. Dagger: efficient and fast RPCs in cloud microservices with near-memory reconfigurable NICs. In *Proceedings of the 26th ACM International Conference on Architectural Support for Programming Languages and Operating Systems* (Virtual, USA) *(ASPLOS '21)*. Association for Computing Machinery, New York, NY, USA, 36–51. https://doi.org/10.1145/3445814.3446696

[20] Shanqing Lin, Ali Ansari, Yuanlong Li, Ayan Chakraborty, Bugra Eryilmaz, Mohammad Alian, and Babak Falsafi. 2025. QFlex 3.0: Fast and Accurate ARM Server Simulation. In *ARM-based General-Purpose Computing: Software-Hardware Co-Optimization for Performance Acceleration*.

[21] Pejman Lotfi-Kamran, Boris Grot, Michael Ferdman, Stavros Volos, Yusuf Onur Koçberber, Javier Picorel, Almutaz Adileh, Djordje Jevdjic, Sachin Idgunji, Emre Ozer, and Babak Falsafi. 2012. Scale-out processors. In *39th International Symposium on Computer Architecture (ISCA 2012), June 9-13, 2012, Portland, OR, USA*. IEEE Computer Society, 500–511. https://doi.org/10.1109/ISCA.2012.6237043

[22] Sarah McClure, Amy Ousterhout, Scott Shenker, and Sylvia Ratnasamy. 2022. Efficient Scheduling Policies for Microsecond-Scale Tasks. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*. USENIX Association, Renton, WA, 1–18. https://www.usenix.org/conference/nsdi22/presentation/mcclure

[23] NVIDIA. 2024. *NVIDIA Grace Hopper Superchip Architecture Whitepaper*. Technical Report. NVIDIA. Accessed: 2025-11-15.

[24] Ali Pahlevan, Javier Picorel, Arash Pourhabibi Zarandi, Davide Rossi, Marina Zapater, Andrea Bartolini, Pablo G. Del Valle, David Atienza, Luca Benini, and Babak Falsafi. 2016. Towards near-threshold server processors. In *2016 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. 7–12.

[25] Hamza Bin Sohail, Mithuna Thottethodi, and T. N. Vijaykumar. 2011. *Dark Silicon is Sub-Optimal and Avoidable*. Technical Report TR-ECE-11-22. Department of Electrical and Computer Engineering, Purdue University. https://docs.lib.purdue.edu/cgi/viewcontent.cgi?article=1435&context=ecetr Technical Report.

[26] Wei Su, Abhishek Dhanotia, Carlos Torres, Jayneel Gandhi, Neha Gholkar, Shobhit Kanaujia, Maxim Naumov, Kalyan Subramanian, Valentin Andrei, Yifan Yuan, and Chunqiang Tang. 2025. DCPerf: An Open-Source, Battle-Tested Performance Benchmark Suite for Datacenter Workloads. In *Proceedings of the 52nd Annual International Symposium on Computer Architecture (ISCA '25)*. Association for Computing Machinery, New York, NY, USA, 1717–1730. https://doi.org/10.1145/3695053.3731411

[27] Michael B Taylor. 2012. Is dark silicon useful? Harnessing the four horsemen of the coming dark silicon apocalypse. In *Proceedings of the 49th annual design automation conference*. 1131–1136.

[28] Thomas F. Wenisch, Roland E. Wunderlich, Michael Ferdman, Anastassia Ailamaki, Babak Falsafi, and James C. Hoe. 2006. SimFlex: Statistical Sampling of Computer System Simulation. *IEEE Micro* 26, 4 (July 2006), 18–31. https://doi.org/10.1109/MM.2006.79