Performance Is the Sole Target: Power Budgeting Assisted Thermal-Aware Processor Floorplanning

Hai Wang, *Member, IEEE*, Zhijie Zhang, and Changyi Qin

***Abstract*—Traditional thermal-aware floorplanning techniques have historically framed layout generation as a multi-objective optimization problem, attempting to simultaneously balance interconnect wire length (delay) and thermal distribution. This research argues that relying on multi-objective weights to balance these phenomena is often suboptimal due to the non-linear, discrete nature of clock-cycle latencies. Instead, the Instructions Per Second (IPS) executed under a thermal constraint can serve as the sole optimization target. By establishing IPS as the singular cost function, the interplay between interconnect delay, power budget, and thermal characteristics is naturally and optimally resolved. A novel power-budget-assisted floorplanning framework, named PIST (Performance Is the Sole Target), is proposed. For any given layout, the algorithm calculates the maximum permissible power budget under a predefined temperature threshold, and translates the power budget into a peak operating frequency. Simultaneously, wire lengths are converted into physical delays and then into cycle-level latencies based on the derived frequency. Using a pre-characterized microarchitectural look-up table, these cycle latencies are mapped to performance degradation and finally form the IPS estimation. PIST can be integrated into an optimization engine like simulated annealing, which generates the floorplan that maximizes such IPS estimation. Evaluated using the GEM5 simulator and the PARSEC benchmark suite, PIST is compared against both classical and state-of-the-art thermal-aware methods. The results demonstrate that optimizing solely for IPS prevents the over-optimization of either delay or temperature, yielding superior performance by maintaining critical path delays within the integer cycle limit while maximizing the allowable power budget and frequency.**

***Index Terms*—floorplanning, power budget, processor design, thermal-aware design, physical design.**

I. Introduction and Motivation

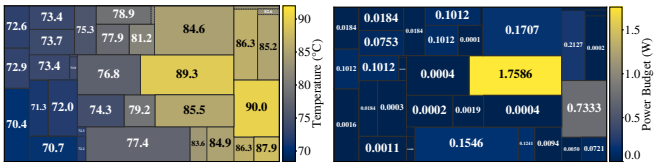
As transistor density increases in the sub-10nm regime, power density has created a severe microarchitectural crisis [1], [2]. The breakdown of Dennard scaling means modern cooling solutions cannot keep up [3], making thermal management a first-class constraint at the earliest stages of microarchitectural design and physical layout [4], [5].

Floorplanning dictates the spatial arrangement of modules on the silicon die, establishing the baseline for performance (interconnect delay) and chip temperature. Traditionally, layout algorithms minimized wire length [6]–[9], but placing active components close together creates thermal hotspots [10]. These hotspots trigger dynamic thermal management (DTM) mechanisms [11], [12], resulting in performance degradation due to scaled-down voltage and frequency.

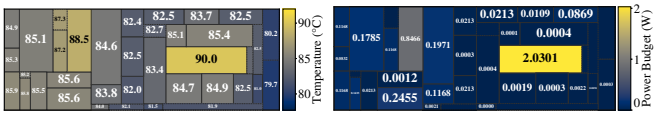
To address this problem, modern methodologies often formulate thermal-aware floorplanning as a multi-objective optimization problem [5], [10], [13]. However, balancing interconnect delay and thermal distribution with scalar weights presents distinct challenges: the arbitrary assignment of scalar weights fails to capture the complex, non-linear relationship between operating frequency, physical wire delay, and the actual computational throughput of the superscalar processor. Minimizing delay creates compact layouts that overheat instantly, demanding a restricted power budget and a low operating frequency. Conversely, prioritizing thermal uniformity spreads components far apart, leading to long global interconnects that incur multi-cycle communication latencies and degrade Instructions Per Cycle (IPC).

H. Wang, Z. Zhang, and C. Qin are with School of Integrated Circuit Science and Engineering, University of Electronic Science and Technology of China, Chengdu, 611731 China.

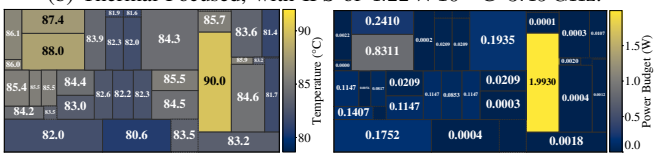
This research is supported by National Natural Science Foundation of China under grant No. 61974018.



(a) Delay-Focused, with IPS of 4.32×10^9 @ 3.31 GHz.



(b) Thermal-Focused, with IPS of 4.22×10^9 @ 3.45 GHz.



(c) IPS-Optimized, with IPS of 4.49×10^9 @ 3.44 GHz.

Fig. 1: Thermal and delay related results running the blackscholes benchmark, with temperature threshold of 90 °C.

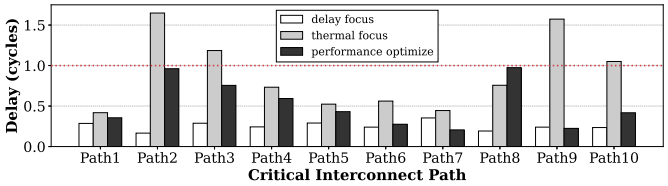


Fig. 2: Delay comparison of the three floorplans.

A motivational example comparing three floorplan topologies for the same processor architecture under a 90 °C thermal threshold illustrates this limitation (Figures 1 and 2):

- Delay-Focused Layout: Generated by multi-objective optimization with heavy weighting on interconnect delay. It creates a layout with uneven temperature distribution and significant hotspots, forcing a low power budget and low clock frequency. While IPC is perfectly preserved due to sub-cycle delays, the low frequency still results in poor global performance measured in Instructions Per Second (IPS).
- Thermal-Focused Layout: Generated by multi-objective optimization prioritizing thermal distribution. It spreads out components, allowing higher power budget and clock frequency. However, extended global interconnects exceed the 1-cycle limit, leading to IPC degradation that outweighs the frequency gains.
- IPS-Optimized Layout: This is the optimal solution that this work seeks. It spreads components enough to boost the power budget and frequency, but keeps them close enough so criti-

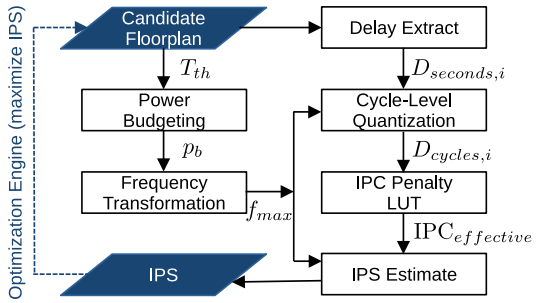


Fig. 3: Flowchart of PIST.

cal path delays remain just under the integer cycle threshold, leading to the best performance in IPS. Linear multi-objective optimizations cannot mathematically identify this delicate, non-linear boundary.

Motivated by the above, this work introduces PIST (Performance Is the Sole Target), a novel floorplanning framework that establishes computational throughput under thermal constraint as the sole optimization goal. By abandoning multi-objective proxy metrics, PIST dynamically calculates power budgets to translate thermal effects and physical wire delays into throughput estimations quantified in IPS. Validated via the GEM5 simulator and PARSEC benchmarks, PIST outperforms both classical and state-of-the-art thermal-aware methodologies in total system performance.

II. The Power Budget Assisted Floorplanning

A. Method Overview

The fundamental idea driving PIST is that the true cost of a floorplan is the performance (measured in IPS) of the corresponding processor under thermal constraint.

The flowchart of PIST is presented in Figure 3. For every candidate floorplan generated by the optimization engine in an optimization loop (for example, in simulated annealing (SA) [6], [10]), the algorithm executes a rapid, five-step analytical sequence:

1) Power Budgeting: Compute the maximum power budget the specific physical topology can sustain under a rigid temperature threshold.
2) Operating Frequency Transformation: Convert the sustainable power budget into a peak operating frequency using technology-specific power models.
3) Physical Delay Extraction: Calculate the point-to-point Manhattan distances of all critical interconnects and derive their physical signal propagation delays in seconds.
4) Cycle-Level Quantization: Translate the physical interconnect delays into discrete clock cycles using the derived operating frequency.
5) IPC Penalty and IPS Estimation: Map the cycle-level latencies to an IPC drop via a pre-characterized microarchitectural Look-Up Table (LUT), and compute the final IPS to serve as the optimization cost.

B. Power Budgeting and Operating Frequency Transformation

The power and thermal dynamics of the processor are evaluated using an equivalent thermal resistance-capacitance (RC) network. At steady state, the temperature vector $T \in \mathbb{R}^n$ of a processor with n functional blocks (FBs) running at frequency f can be expressed as [14], [15]

$$T = RP(f) + T_{amb} \tag{1}$$

where T_{amb} represents the ambient temperature vector, $R \in \mathbb{R}^{n\times n}$ is the thermal resistance matrix, $P(f) = [p_1(f), p_2(f), \ldots, p_n(f)]$ is power vector with $p_i(f)$ representing the power of the ith FB.

Because the power consumption of FBs in a processor is highly correlated [16], the power of FBs can be written as a function of a base power, denoted as $p(f)$, at an operating frequency f. For example, using a linear function, the power of the ith FB is written as:

$$p_i(f) = \alpha_i p(f) + \beta_i \tag{2}$$

where α_i and β_i are fitting parameters. Then the system-wide power vector can be expressed as $P(f) = Ap(f) + B$, where $A = [\alpha_1, \alpha_2, \ldots, \alpha_n]^T$ and $B = [\beta_1, \beta_2, \ldots, \beta_n]^T$, the thermal model (1) becomes

$$T = R(Ap(f) + B) + T_{amb} \tag{3}$$

The budget of the base power p_b is readily computed by solving the following optimization problem:

$$\max\{p_b \in \mathbb{R} \mid R(Ap_b + B) + T_{amb} \preceq T_{th}\} \tag{4}$$

where T_{th} is the temperature threshold vector.

Finally, the operational frequency f_{max} of the processor is algebraically estimated from the maximum sustainable base power budget using the pre-calibrated power model $p(f_{max}) = p_b$. This derived peak frequency is then used in subsequent evaluation steps including the mapping of physical delays to cycle-level latencies and the IPS estimation.

C. Cycle-Level Delay and Performance Estimation

Simultaneously, the physical properties of the candidate layout must be evaluated. The spatial coordinates of the communicating microarchitectural blocks define the global interconnect lengths. For each critical path i, the physical wire length L_i is determined via Manhattan distance. With L_i, the physical signal propagation delay $D_{seconds,i}$ is calculated using an Elmore delay model. Practically, to account for the post-placement routing detours and buffer insertion, $D_{seconds,i}$ can be artificially inflated by an estimated margin.

Crucially, the physical delay must be translated into the temporal domain of the microprocessor. Using the maximum sustainable frequency f_{max} derived in Section II-B, the delay in seconds is quantized into discrete clock cycles:

$$D_{cycles,i} = \lceil D_{seconds,i} \times f_{max} \rceil \tag{5}$$

This formulation captures the step-function penalty of wire elongation.

The final phase evaluates how these cycle-level latencies impact the execution pipeline. Because invoking a full-system, cycle-accurate architectural simulator (such as GEM5) inside the optimization loop is computationally prohibitive, the framework employs a pre-characterized Look-Up Table (LUT) similar to [10].

Prior to layout optimization, exhaustive offline profiling is conducted using the GEM5 simulator across different benchmarks. The latencies of the most critical microarchitectural interconnects are systematically varied from 0 to 5 cycles. The resulting performance degradation is captured and averaged across benchmarks, creating an LUT that maps any vector of interconnect cycle delays to an effective Instructions Per Cycle (IPC) drop.

During the optimization loop, the algorithm queries this LUT using the computed D_{cycles} vector to instantaneously retrieve the predicted $\text{IPC}_{effective}$ for the candidate floorplan. The ultimate optimization cost metric is then calculated as the total Instructions Per Second:

$$\text{IPS} = f_{max} \times \text{IPC}_{effective} \tag{6}$$

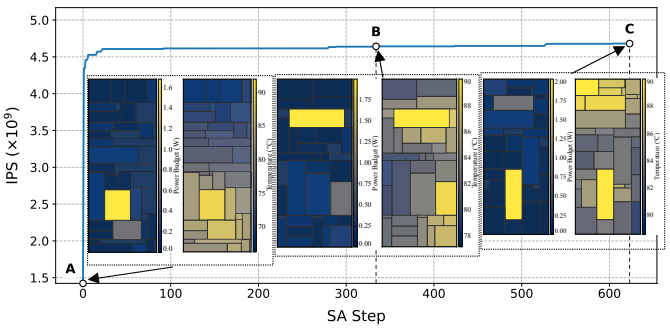


Fig. 4: The estimated IPS, power budget map, and thermal map of the floorplan during the simulated annealing process of PIST.

The optimization engine uses this IPS value to form the cost function, probabilistically accepting or rejecting perturbed topologies to maximize the true computational throughput.

It is critical to note that the PIST framework implicitly performs timing-driven critical-path optimization, because the importance of each critical interconnect is revealed by its IPC drop in the LUT.

In addition, such IPC drops utilized during optimization represent the average computational efficiency drop across the diverse benchmark suite. This statistical formulation is intentional: if the drops were constructed using a single application profile, the resulting floorplan would perform optimally for that specific application but badly on general-purpose software workloads.

III. Experimental Results

PIST is compatible with many popular optimization frameworks including Simulated Annealing (SA) and Genetic Algorithm (GA). In this experiment, we implemented PIST in the SA framework to demonstrate its performance.

The microarchitectural performance profiling and LUT generation were executed using the GEM5 simulator [17], with the widely used PARSEC benchmark suite [18]. The simulated system models an O3 X86 processor architecture.

The thermal properties of the floorplans were evaluated using HotSpot [19]. The ambient temperature was set to $45\,^{\circ}\mathrm{C}$, and the thermal threshold was clamped at $90\,^{\circ}\mathrm{C}$ to simulate real-world DTM trigger points.

PIST is benchmarked against both classical and state-of-the-art methodologies:

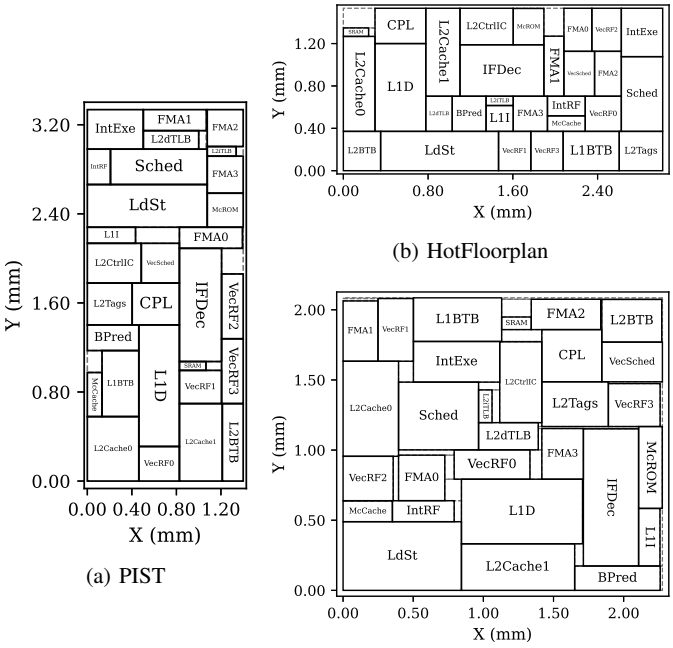
- HotFloorplan [10]: A standard Simulated Annealing framework utilizing a multi-objective cost function weighted towards minimizing peak temperatures and delays of critical paths, reflecting classical thermal-aware methodologies.
- Bayesian Optimization (BO) [5]: A state-of-the-art methodology that employs a Gaussian Process surrogate model and specialized permutation kernels to minimize the thermal gradient and equalize heat across the die.

The same area utilization constraint of 94% is applied to all methods.

The following subsections detail the convergence behavior, structural topologies, and quantitative performance superiority of the framework.

A. Optimization Convergence Analysis

Figure 4 illustrates the dynamic convergence of the simulated annealing process of PIST, mapping the estimated IPS, power budgets, and thermal distribution across the iteration timeline.



(a) PIST

(b) HotFloorplan

(c) BO

Fig. 5: Floorplans generated by different methods.

At the beginning of the SA process, the randomly generated topologies exhibit extreme thermal hotspots and inefficient wire routing. The uneven temperature distribution forces the model to severely curtail the allowed power budget. Consequently, the estimated IPS is depressed.

As the optimization traverses through the middle steps, the algorithm begins to disperse the heavily utilized functional units. This targeted diffusion acts to increase the overall thermal conductance to the heat sink, raising the sustainable power budget. Crucially, the estimated IPS curve rises as the power budget increases and critical wire lengths shorten.

By the final optimization stages (around step 620), the power budget has been raised further with improved thermal distribution, while the layout still ensures that interconnect distances remain short enough to prevent multi-cycle latencies, which leads to an even higher estimated IPS.

From Figure 4, we also see that the estimated IPS (around 4.7×10^9) of the final layout lies within the range of the actual simulation results of different benchmarks shown later in Figure 6a, proving the LUT-based IPS estimation works well for general-purpose execution.

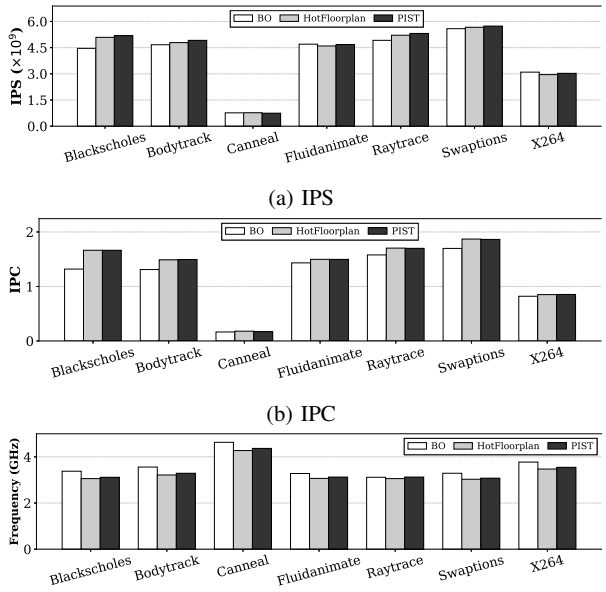
The floorplans generated by PIST, HotFloorplan, and BO are given in Figure 5.

B. Performance Superiority

The true metric of success is the resultant computational throughput. Figure 6 details the performance comparison of the floorplans generated by the three methods across different PARSEC benchmarks.

PIST achieves the highest overall IPS across the benchmark suite (Figure 6a). This superiority is achieved through an optimal synthesis of both frequency and IPC:

- IPC Preservation (Figure 6b): Both PIST and HotFloorplan achieve similarly high IPC values. Because both methods restrict critical path lengths, they do not suffer from the pipeline stalling and memory latency penalties inherent in dispersed designs. In



(a) IPS

(b) IPC

(c) Frequency

Fig. 6: Comparison of the new method and the traditional methods on PARSEC benchmark suite.

contrast, the Bayesian Optimization layout suffers significant IPC drop due to the extended network-on-chip routing latencies caused by its homogenized thermal placement.

- Frequency Maximization (Figure 6c): Both PIST and Bayesian Optimization sustain similar, high operational frequencies. By adequately managing the thermal density, both methods unlock a large power budget. HotFloorplan tends to value delay more than temperature in the multi-objective optimization, resulting in a lower operating frequency to prevent breaching the $90\,^{\circ}\mathrm{C}$ limit.

The IPS-driven PIST framework succeeds because it captures the best elements of both the high IPC of delay-focused routing and the high frequency of thermal-focused optimization, yielding the highest computational throughput.

C. Thermal and Critical Path Delay Dynamics

To explicitly validate why PIST performs best, we analyze the micro-level thermal and delay dynamics under the blackscholes benchmark, utilizing the power and thermal maps in Figure 7 and the critical path delay profiles in Figure 8.

HotFloorplan achieves ultra-short physical delays, with all critical paths completing in a fraction of a cycle. However, this is an architectural over-optimization: a signal resolving in 0.3 cycles provides no IPC benefit over a signal resolving in 0.9 cycles, yet achieving 0.3 cycles exacts a thermal penalty as visualized in Figure 7.

Bayesian Optimization pushes the physical wires so far to achieve its thermal uniformity that multiple critical paths exceed the 1-cycle duration at the operating frequency (Figure 8). These multi-cycle delays disrupt out-of-order instruction scheduling and inflate the Reorder Buffer occupancy, devastating the IPC.

PIST monitors this discrete boundary automatically during the optimization process. It intentionally elongates wire delays compared to HotFloorplan—sacrificing unnecessary physical proximity to harvest thermal headroom (Figure 7)—but rigidly halts this topological spreading just before any path breaches the integer cycle threshold (Figure 8). This perfectly calculated edge is completely invisible to multi-objective algebraic formulations, and is solely discoverable when the layout is evaluated via raw, cycle-quantized IPS.

D. Area Utilization Analysis

The primary experiment applied a 94% area utilization constraint across all methods. However, relaxing this constraint may improve heat dissipation through geographic separation.

When area utilization constraint is completely removed, PIST naturally converges on a layout of 76% area utilization, with performance improved by 3% on average compared with the original floorplan in Figure 5a. This indicates that PIST leverages this additional geographic space to dissipate thermal hotspots, and prevents the elongated critical wires from breaching the discrete integer cycle limits to degrade computational efficiency.

E. Runtime Analysis

The integration of the thermal resistance matrix computation does introduce computational overhead compared to purely geometric placements. However, tested on a laptop with Intel Core i7-13650HX, the IPS estimation of PIST for each candidate floorplan takes merely $28\,\mathrm{ms}$ on average.

Consequently, simulated annealing with PIST executes entirely for around 1.5 hours with 620 SA steps. It exhibits a longer runtime than the highly abstracted BO approach which converges faster (around 400 steps consuming 55 minutes), but remains comparable to HotFloorplan.

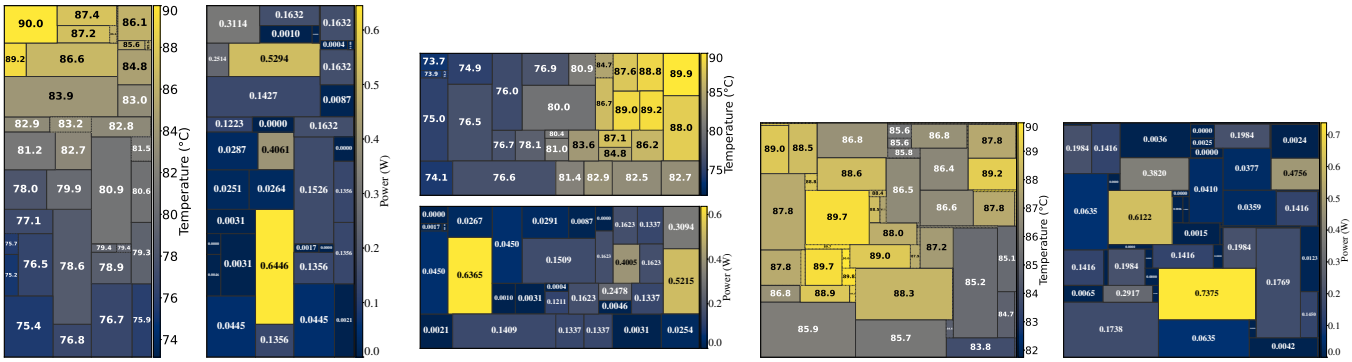
Given that floorplanning is executed during the design phase, a less than 2-hour runtime on a laptop is considered minimal, particularly when offset by the profound gains in finalized silicon performance and energy efficiency.

IV. Future Work

In this brief, we have, for the first time, demonstrated how to perform thermal-aware processor floorplanning using a single-objective, performance-driven optimization framework. Recent research focuses heavily on advanced learning-based models to accelerate the optimization process [20], [21] and 2.5D/3D package-level floorplanning/placement [22], [23]. As a result, we plan to develop a machine learning-based optimization engine, such as operator learning models, to accelerate the convergence of PIST. Additionally, we will extend this framework to package-level systems, specifically 3-D ICs with multi-layer co-optimization and Through-Silicon Via (TSV) placement, to effectively manage lateral thermal blockages and localized hotspots in stacked architectures.

V. Conclusion

This paper demonstrates that traditional multi-objective optimization based physical layout algorithms inherently over-optimize either thermal distribution at the cost of IPC, or physical density at the cost of operating frequency. By establishing Instructions Per Second (IPS) under thermal constraint as the singular optimization target, the PIST floorplanning framework resolves the thermal-delay contradiction. By iteratively computing the maximum sustainable power budget via thermal matrices, deriving the peak frequency, and precisely mapping physical wire delays to cycle-level IPC penalties using pre-characterized simulations, the algorithm successfully handles the discontinuous performance landscape. Empirical validation utilizing the GEM5 simulator and PARSEC workloads confirms that the IPS-driven methodology outperforms both classical Simulated Annealing and advanced Bayesian Optimization. It meticulously harvests thermal headroom to maximize clock frequency while rigidly protecting



(a) PIST (b) HotFloorplan (c) BO

Fig. 7: Power and thermal maps of floorplans generated by different methods running the blackscholes benchmark, with the highest temperature clamped at 90 °C.

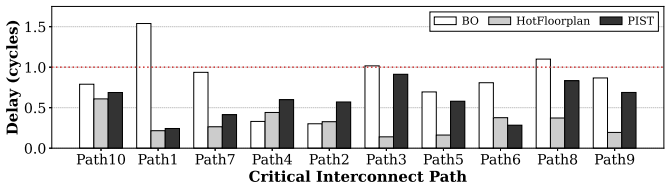


Fig. 8: Delay comparison of the three floorplans running the blackscholes benchmark.

critical path communication from multi-cycle degradation, ultimately extracting the maximum possible performance from thermally constrained silicon.

REFERENCES

[1] J. H. Yahya, J. S. Kim, A. G. Yağlıkçı, J. Park, E. Rotem, Y. Sazeides, and O. Mutlu, "DarkGates: A hybrid power-gating architecture to mitigate the performance impact of dark-silicon in high performance processors," in *Proc. IEEE Int. Symp. on High-Performance Computer Architecture (HPCA)*, April 2022, pp. 1170–1183.

[2] P. Dangi, T. K. Bandara, S. Sheikhpour, T. Mitra, and L. Eeckhout, "Sustainable hardware specialization," in *Proc. Int. Conf. on Computer Aided Design (ICCAD)*, October 2024.

[3] P. Subrahmanyam, S. Basak, V. P. Sugumar, Y.-F. Pang, A. Krishnamoorthy, and M. Bianco, "Automated gimbaling of conical nozzles for jet vectoring impingement cooling to mitigate workload dependent hotspots in high power density silicon," in *Proc. Thermal and Thermomechanical Phenomena in Electronics Systems*, 2024.

[4] Z. Seifoori, B. Omidi, and H. Asadi, "PERA: Power-efficient routing architecture for SRAM-based FPGAs in dark silicon era," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 12, pp. 2075–2088, December 2023.

[5] M. Molter, R. Kumar, S. Koller, O. W. Bhatti, N. Ambasana, and E. Rosenbaum, "Thermal-aware SoC macro placement and multi-chip module design optimization with bayesian optimization," in *Proc. Electronic Components and Technology Conference (ECTC)*, May 2023.

[6] J. Cong, A. Jagannathan, G. Reinman, and M. Romesis, "Microarchitecture evaluation with physical planning," in *Proc. Design Automation Conf. (DAC)*, June 2003, pp. 32–35.

[7] S. Mohapatra, S. K. Vendra, and M. Chrzanowska-Jeske, "Fast buffer count estimation in 3D IC floorplanning," *IEEE Trans. on Circuits and Systems II: Express Briefs*, vol. 68, no. 1, pp. 271–275, January 2021.

[8] Y. Liu, H. Zhou, J. Wang, F. Yang, X. Zeng, and L. Shang, "Hierarchical graph learning-based floorplanning with Dirichlet boundary conditions," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 32, no. 5, pp. 810–822, May 2024.

[9] W. Guan, X. Tang, H. Lu, J. Tan, J. Wang, Y. Zhang, and Y. Zhang, "LSTM-characterized approach for chip floorplanning: Leveraging HyperGCN and DRQN," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 44, no. 2, pp. 709–722, February 2025.

[10] K. Sankaranarayanan, S. Velusamy, M. Stan, and K. Skadron, "A case for thermal-aware floorplanning at the microarchitectural level," *Journal of Instruction-Level Parallelism*, vol. 7, pp. 1–16, 2005.

[11] H. Wang, L. Hu, X. Guo, Y. Nie, and H. Tang, "Compact piecewise linear model based temperature control of multi-core systems considering leakage power," *IEEE Trans. on Industrial Informatics*, vol. 16, no. 12, pp. 7556–7565, December 2020.

[12] K.-C. Chen, Y.-H. Liao, C.-T. Chen, and L.-Q. Wang, "Adaptive machine learning-based proactive thermal management for NoC systems," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 8, pp. 1114–1127, August 2023.

[13] W. Guan, X. Tang, H. Lu, Y. Zhang, and Y. Zhang, "Thermal-aware fixed-outline 3-D IC floorplanning: An end-to-end learning-based approach," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 12, pp. 1882–1895, December 2023.

[14] H. Wang, D. Tang, M. Zhang, S. X.-D. Tan, C. Zhang, H. Tang, and Y. Yuan, "GDP: A greedy based dynamic power budgeting method for multi/many-core systems in dark silicon," *IEEE Trans. on Computers*, vol. 68, no. 4, pp. 526–541, April 2019.

[15] S. Pagani, H. Khdr, J.-J. Chen, M. Shafique, M. Li, and J. Henkel, "Thermal safe power (TSP): Efficient power budgeting for heterogeneous manycore systems in dark silicon," *IEEE Trans. on Computers*, vol. 66, no. 1, pp. 147–162, January 2017.

[16] H. Wang, S. X.-D. Tan, G. Liao, R. Quintanilla, and A. Gupta, "Full-chip runtime error-tolerant thermal estimation and prediction for practical thermal management," in *Proc. Int. Conf. on Computer Aided Design (ICCAD)*, November 2011, pp. 716–723.

[17] N. Binkert, B. Beckmann, G. Black, S. K. Reinhardt, A. Saidi, A. Basu, J. Hestness, D. R. Hower, T. Krishna, S. Sardashti, R. Sen, K. Sewell, M. Shoaib, N. Vaish, M. D. Hill, and D. A. Wood, "The gem5 simulator," *ACM SIGARCH Comput. Archit. News*, vol. 39, no. 2, pp. 1–7, May 2011.

[18] C. Bienia, "Benchmarking modern multiprocessors," Ph.D. dissertation, Princeton University, January 2011.

[19] W. Huang, S. Ghosh, S. Velusamy, K. Sankaranarayanan, K. Skadron, and M. R. Stan, "HotSpot: A compact thermal modeling methodology for early-stage VLSI design," *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, vol. 14, no. 5, pp. 501–513, May 2006.

[20] Z. Liu, Y. Li, J. Hu, X. Yu, S. Shiau, X. Ai, Z. Zeng, and Z. Zhang, "DeepOHeat: Operator learning-based ultra-fast thermal simulation in 3D-IC design," in *Proc. Design Automation Conf. (DAC)*, July 2023.

[21] X. Liu, S. Liu, B. Yu, S. Chen, and Q. Xu, "ThePlace: Thermal-aware placement with operator learning-based ultra-fast simulator," in *Proc. Asia South Pacific Design Automation Conf. (ASP-DAC)*, January 2025, pp. 900–906.

[22] H.-W. Chiou, J.-H. Jiang, Y.-T. Chang, Y.-M. Lee, and C.-W. Pan, "Chiplet placement for 2.5D IC with sequence pair based tree and thermal consideration," in *Proc. Asia South Pacific Design Automation Conf. (ASP-DAC)*, January 2023, pp. 7–12.

[23] Q. Wang, X. Li, T. Jia, Y. Lin, R. Wang, and R. Huang, "ATPlace2.5D: Analytical thermal-aware chiplet placement framework for large-scale 2.5D-IC," in *Proc. Int. Conf. on Computer Aided Design (ICCAD)*, October 2024.