TAP-2.5D: A Thermally-Aware Chiplet Placement Methodology for 2.5D Systems

Yenai Ma, Leila Delshadtehrani, Cansu Demirkiran, José L. Abellán*, Ajay Joshi
*Boston University, *Catholic University of Murcia*
{yenai, delshad, cansu, joshi}@bu.edu, jlabellan@ucam.edu

***Abstract*—Heterogeneous systems are commonly used today to sustain the historic benefits we have achieved through technology scaling. 2.5D integration technology provides a cost-effective solution for designing heterogeneous systems. The traditional physical design of a 2.5D heterogeneous system closely packs the chiplets to minimize wirelength, but this leads to a thermally-inefficient design. We propose TAP-2.5D: the first open-source network routing and thermally-aware chiplet placement methodology for heterogeneous 2.5D systems. TAP-2.5D strategically inserts spacing between chiplets to jointly minimize the temperature and total wirelength, and in turn, increases the thermal design power envelope of the overall system. We present three case studies demonstrating the usage and efficacy of TAP-2.5D.**

***Index Terms*—heterogeneous 2.5D systems, thermally-aware placement, inter-chiplet network**

I. Introduction

2.5D integration technology enables the design of heterogeneous System-in-Package (SiP) consisting of multiple different chiplets (CPU, GPU, memory, etc.) fabricated using different technologies and processes [1], [2]. 2.5D integration places chiplets side by side on a silicon interposer, which enables inter-chiplet communication using links whose properties are comparable to that of on-chip links. Hence, a SiP design achieves higher performance and lower energy consumption than traditional monolithic (2D) based systems [3]. Moreover, compared to a printed circuit board approach, a 2.5D integrated SiP shrinks the system footprint significantly. Several commercial products such as AMD Fiji [4], Nvidia Tesla [5], and Intel Foveros [6] are already using 2.5D technology.

There are multiple options for 2.5D integration technology, including active interposer and passive interposer[1]. An active interposer is effectively a large carrier chip containing transistors. It is expensive as it requires front-end-of-line (FEOL) process and suffers from yield loss when the area is large. A passive interposer is transistor-free, and so it can be fabricated using a back-end-of-line (BEOL) process and inherently has higher yield [10] and is much cheaper to fabricate [10]). Due to its cost effectiveness and placement flexibility, we focus on passive interposer based 2.5D integration (see Fig. 1).

One of the design challenges in these heterogeneous 2.5D integrated systems is the physical design of the inter-chiplet network. Here, the physical design specifies *how to place the chiplets and how to efficiently route the wires between the chiplets for a given logical topology*. We want to provide the required connectivity between the chiplets, while minimizing area and cost, maximizing performance, and avoiding thermal-based failures. Traditionally, the physical floorplanning of monolithic chips focuses on reducing the total wirelength connecting the macrocells and minimizing the area [11]. This strategy can be adapted and applied to heterogeneous 2.5D systems. However, this ends up with closely packed chiplets, which likely suffer from thermal-based failure if the chiplets have high power densities. To prevent overheating, we have to either apply a more advanced but expensive cooling technology [12], or degrade system performance by turning off some chiplets or lowering the operating frequency of parts of the chiplets [13].

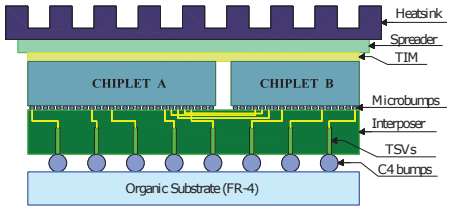


Fig. 1: Cross-section view of a 2.5D system with passive interposer – Epoxy resin is often used to underfill the spacing between C4 bumps, between microbumps, and between chiplets.

In this paper, unlike state-of-the-art methodologies (further details in Section II) that output compact placement of chiplets, we propose to strategically insert spacing between the different chiplets in an heterogeneous 2.5D system to lower temperature while minimizing inter-chiplet wirelength[2]. This new physical design methodology is called TAP-2.5D. TAP-2.5D implements a Simulated Annealing (SA)-based thermally-aware placer that relies on a new placement description data structure called *Occupation Chiplet Matrix (OCM)* (further details in Section III-C1) to explore the physical design space in order to find an inter-chiplet network routing solution, that jointly minimizes the peak operating temperature of the overall system and the total inter-chiplet network wirelength. Besides, TAP-2.5D is able to increase the TDP envelope without using any advanced and costly active cooling methods. This increase in TDP envelope allows higher power budget, which can be used to improve performance. The main contributions of our paper are:

- We propose TAP-2.5D, to the best of our knowledge the first open-source (https://github.com/bu-icsg/TAP-2.5D)

[1]Embedded Multi-die Interconnect Bridge (EMIB) [7] has been recently proposed to connect adjacent chiplets for die-to-die communication. Although, an EMIB-based approach has lower silicon cost than a silicon interposer-based approach [7], we do not explore this technology because EMIB has limited die-to-die connections per layer [8], can only hook up adjacent chiplets, and also has higher complexity in the manufacturing of organic substrates [9].

[2]Any loss of performance from longer wirelength can be recovered through the increased Thermal Design Power (TDP) budget due to lower temperature.

thermally-aware chiplet placement methodology for physical design of inter-chiplet networks targeting passive interposer-based heterogeneous 2.5D systems.
- To prove the efficacy of our approach, we apply TAP-2.5D to three heterogeneous 2.5D systems. For a conceptual Multi-GPU System, TAP-2.5D reduces the operating temperature by 4^oC at the cost of 10% increase in wirelength. For a CPU-DRAM System [14], TAP-2.5D lowers an infeasible high temperature of 113.54^oC by 20^oC, and improves TDP by $150W$. For an existing Huawei Ascend 910 System [15] that is already operating below a critical temperature of 85^oC, we get a placement solution similar to the commercial system as TAP-2.5D focuses on wirelength minimization.

II. Related Work

There have been many works on the design and evaluation of heterogeneous 2.5D systems in recent years. Kim *et al.* [16] present a highly-integrated design flow to build and simulate heterogeneous 2.5D systems. Vijayaraghavan *et al.* [17] present a vision for exascale system architecture consisting of CPUs, GPUs and HBMs, and aggressively use die-stacking and chiplet technologies. Ebrahimi *et al.* [18] propose an independent NoC die for 2.5D heterogeneous manycore systems. Yin *et al.* [19] propose a modular methodology for deadlock-free routing in 2.5D systems. Moreover, there are commercial heterogeneous 2.5D systems such as AMD Fiji [4], Nvidia Tesla [5], and Intel Foveros [6]. All these works typically place the chiplets next to each other on an interposer. This way they benefit from low communication latency (due to short inter-chiplet links) and low manufacturing cost (due to small interposer sizes). However, as we show in this paper, this strategy can lead to bad designs for systems containing high power-density chiplets.

Several works have focused on determining the physical floorplan for networks in monolithic chips. Murata *et al.* [20] propose Sequence Pair to represent a rectangular block packing and use SA to minimize wirelength and area for monolithic chips. Lin *et al.* [21] represent a P-admissible floorplan using Transitive Closure Graph (TCG) and develop an SA-based algorithm. Guo *et al.* [22] propose O-tree as the representation for left and bottom compacted placement and use a deterministic floorplan algorithm. Chen *et al.* [11] propose B*-tree and a fast-SA algorithm to search the solution space. One can simply extend these tools to solve the floorplanning problem for 2.5D systems. However, the admissible placement assumption of all these data structures and methods, where the chiplets are packed closely and cannot move in left and down directions, do not hold for 2.5D systems. In 2.5D systems, we can strategically insert spacing between chiplets to improve heat dissipation. Our work, TAP-2.5D, uses a new OCM data structure (more details in Section III-C1) to represent an unrestricted placement to search for a solution that minimizes both wirelength and operating temperature.

In addition to the traditional design objectives, such as minimizing area and wirelength, many recent floorplanning works consider the thermal aspect. Healy *et al.* [23] present a multi-objective microarchitectural floorplanning algorithm for 2D and 3D systems to achieve both high performance and thermal reliability. Cong *et al.* [24] propose a thermal-driven 3D floorplanning algorithm. The above two works are still limited to compact placement, which cannot be applied to 2.5D systems to leverage the placement flexibility with a larger solution space. Eris *et al.* [13] leverage 2.5D integration technology to strategically insert spacing between chiplets to lower system temperature and reclaim dark silicon. Coskun *et al.* [25], [26] propose a cross-layer (i.e., logical, physical and circuit layers are considered) co-optimization methodology for designing networks in 2.5D systems[3]. These works are restricted to 16-chiplet homogeneous 2.5D systems, where all chiplets are identical and square-shaped, and it covers a limited solution space with matrix-style chiplet placement and symmetry assumptions. To determine the thermally-aware placement for heterogeneous 2.5D systems, our work considers various chiplet counts, chiplet sizes, chiplet shapes, and non-uniform connectivity and bandwidth requirements, and searches for an unrestricted chiplet placement solution with intelligently inserted spacing.

III. TAP-2.5D Methodology

TAP-2.5D is aimed to find an inter-chiplet network routing solution for heterogeneous 2.5D systems that, given a logical inter-chiplet network topology, jointly minimizes the peak operating temperature of the overall system (Section III-A) and the total inter-chiplet network wirelength (Section III-B) following a SA-based thermally-aware placer (Section III-C).

A. Thermal Evaluation

Our thermal simulation takes the chiplet placement from the thermally-aware placer described in Section III-C, and uses the 2.5D system configuration (including chiplet widths and heights, power profiles, system layer descriptions, and material properties) to evaluate the operating temperature. We use an extension [27] of HotSpot that provides detailed heterogeneous 3D modeling features, which supports heterogeneous materials in each modeling layer. To model our 2.5D system, we stack six modeling layers. From the bottom up, as illustrated in Fig. 1, the layers are organic substrate, C4 bump layer, silicon interposer, microbump layer, chiplet layer, and Thermal Interface Material (TIM). We use a separate floorplan for each layer to describe the placement and materials. Our 2.5D system model uses the properties (such as layer thickness, materials, dimensions of bumps, and TSVs) of real systems [28], [29]. Besides, we use a realistic air-forced heatsink as the cooling technique. Following the HotSpot default conventions, we set the ambient temperature to 45^oC, the grid model resolution to 64×64, the heat spreader edge size to be $2\times$ the interposer edge size, and the heatsink edge size to be $2\times$ the spreader edge size. To keep the heat transfer coefficient consistent across all simulations, we adjust the convective resistance of the heatsink. The runtime for each HotSpot simulation is 23 seconds on average.

[3]TAP-2.5D focuses on optimizing the physical layer. However, it can be integrated with a cross-layer methodology similar to Coskun et al. [25], [26] for design and optimization of networks in heterogeneous 2.5D systems. This is part of our future work.

TABLE I: Notations.

Notation	Meaning
C, P, N	Set of chiplets, set of pin clumps, and set of nets, respectively.
c, i, j	Index of a chiplet $\in C$.
p, l, k	Index of a pin clump $\in P$.
n	A net $\in N$.
d_{iljk}	Manhattan Distance from pin clump l on chiplet i to pin clump k on chiplet j. Note that $d_{iljk} = d_{jkil}$.
f^n_{iljk}	Flow variable. Number of wires from pin clump l of chiplet i to pin clump k of chiplet j that belong to net n.
X_c, Y_c	Center x- and y-coordinates for chiplet c.
x_p, y_p	x- and y-offsets from center point of the chiplet for pin clump p.
s_n, t_n	Source chiplet and sink chiplet of net n, respectively.
R_{ij}	Input requirement on the wire count between chiplet i and chiplet j.
P^{max}_{il}	Microbump capacity for a pin clump l on chiplet i.
w_i, h_i	Width and height of chiplet i.
w_{gap}	Minimum spacing between two chiplets: $100\mu m$ [2].
w_{int}	Edge length of interposer, $w_{int} \leq 50mm$ [25].

B. Routing Optimization

The objective of our routing tool is to find a routing solution that minimizes the total wirelength of the inter-chiplet network. We frame the delivery of required number of wires between chiplets as a multi-commodity flow, which is NP-hard, so we formulate it as a Mixed-integer linear programming (MILP) solver that was inspired by the one used by Coskun et al. [25] that targets less challenging homogeneous chiplets[4]. Also, to limit the problem size, we group the microbumps along the chiplet periphery into pin clumps[5]. The inputs of the MILP solver are the chiplet placement from our thermally-aware placer (Section III-C), the estimated microbump resources for inter-chiplet communication, and the inter-chiplet connectivity and bandwidth requirements of the 2.5D system. The MILP solver outputs the optimal routing solution and the corresponding total wirelength. We formulate our MILP solver as follows (the notations are listed in Table I):

$$\textbf{Minimize:} \quad \sum_{i\in C, l\in P, j\in C, k\in P, n\in N} d_{iljk} \cdot f^n_{iljk} \tag{1}$$

Subject to:

$$d_{iljk} = |X_i + x_l - X_j - x_k| + |Y_i + y_l - Y_j - y_k| \tag{2}$$

$$f^n_{iljk} \geq 0, \qquad \forall i \in C, l \in P, j \in C, k \in P, n \in N \tag{3}$$

$$\sum_{l\in P, j\in C, k\in P} f^n_{iljk} - \sum_{l\in P, j\in C, k\in P} f^n_{jkil} = \begin{cases} R_{s_n t_n}, & \text{if } i = s_n, \forall n \in N \\ -R_{s_n t_n}, & \text{if } i = t_n, \forall n \in N \\ 0, \forall i \neq s_n || t_n, \forall n \in N \end{cases} \tag{4}$$

$$f^n_{jks_n l} = 0, \qquad \forall n \in N, \forall l \in P, \forall j \in C, \forall k \in P \tag{5}$$

$$f^n_{t_n ljk} = 0, \qquad \forall n \in N, \forall l \in P, \forall j \in C, \forall k \in P \tag{6}$$

$$\sum_{j\in C, k\in P, n\in N} f^n_{iljk} + \sum_{j\in C, k\in P, n\in N} f^n_{jkil} \leq P^{max}_{il}, \quad \forall i \in C, l \in P \tag{7}$$

$$\sum_{i\in C, l\in P, j\in C, k\in P} f^n_{iljk} \leq R_{s_n t_n} \tag{8}$$

$$\sum_{i\in C, l\in P, j\in C, k\in P} f^n_{iljk} \leq 2 \cdot R_{s_n t_n} - \sum_{l\in P, k\in P} f^n_{s_n l t_n k} \tag{9}$$

Eqn. (1) is the objective function for the MILP, which sums up the total length of the wires. The route distance d_{iljk} is calculated using Eqn. (2). Eqn. (3) ensures that the flow variable f^n_{iljk} is non-negative. Eqn. (4) guarantees the sum of all flows for a net n, over all pin clumps from source chiplet s_n to sink chiplet t_n, meets the bandwidth requirement, and also assures that the net flow (total outgoing flows f^n_{iljk} minus total incoming flows f^n_{jkil}) is 0 for all other (non-source, non-sink) chiplets for the given net. Eqn. (5) makes sure that there is no input flow (for net n) for any pin clump in the source chiplet s_n from any other chiplet's pin clump. Similarly, Eqn. (6) ascertains no output flow (for net n) for any pin clump in the sink chiplet t_n to any other chiplet's pin clump. Eqn. (7) ensures that all routes have available pins. Eqn. (8) constrains the sum of all flows for a net n within the bandwidth requirement between the source and sink chiplets of the net.

In addition to the repeaterless non-pipelined inter-chiplet links, we consider *gas-station* links [25] that use transistors on an intermediate chiplet to "refuel" the signals and thus enable pipelining in passive interposers. For 2-stage *gas-station* links, we replace Eqn. (8) with Eqn. (9), where the net connects s_n and t_n through at most one other chiplet. Our MILP solver is implemented with IBM ILOG CPLEX v12.8 Python API. The average runtime for each routing optimization is 5 seconds.

C. Thermally-Aware Placement Algorithm

We develop an SA-based algorithm to determine the thermally-aware chiplet placement for heterogeneous 2.5D systems with the provided inter-chiplet connectivity at the logical level. Our methodology faces two key challenges. First, we strategically insert spacing between chiplets to improve heat dissipation. So we cannot use the state-of-the-art floorplan representations, such as Sequence Pair [20], TCG [21], O-tree [22], and B*-tree [11], as these representations assume compact placement. Second, the thermal evaluation and routing optimization processes for each chiplet placement take approximately 30 seconds. Within an acceptable simulation time, our methodology has to find a satisfactory solution with limited steps. We present below the details of the key components of our algorithm, and explain how we overcome the above mentioned challenges.

1) Placement description: To represent unrestricted placements, we use x and y coordinates of the center points of chiplets, together with the widths and heights of the chiplets. To avoid an infinite solution space, we divide the interposer into a discrete grid called *Occupation Chiplet Matrix* (OCM) that is logically shown in Figure 2(a), and we assume that the center of a chiplet can only be placed on the intersection nodes of the grid. We assume $1mm$ granularity for the grid to place the centers of chiplets (the widths and heights of the chiplets can be any value), which for a finite amount of time provides a good balance between the solution space and the solution quality. A valid chiplet placement has no overlap between any pair of chiplets and ensures $0.1mm$ minimum gap between chiplets [28] (Eqn. (10)). It is also necessary for a chiplet to be completely on the interposer (Eqn. (11)).

$$max\{(X_i - \frac{w_i}{2}) - (X_j + \frac{w_j}{2}), (X_j - \frac{w_j}{2}) - (X_i + \frac{w_i}{2}), (Y_i - \frac{h_i}{2}) - (Y_j + \frac{h_j}{2}), (Y_j - \frac{h_j}{2}) - (Y_i + \frac{h_i}{2})\} \geq w_{gap}, \qquad \forall i \in C, \forall j \in C, i \neq j \tag{10}$$

$$\frac{w_i}{2} \leq X_i \leq w_{int} - \frac{w_i}{2}, \quad \frac{h_i}{2} \leq Y_i \leq w_{int} - \frac{h_i}{2}, \qquad \forall i \in C \tag{11}$$

[4]In TAP-2.5D, we also need to account for non-uniform bandwidth between chiplets, and the different dimensions and asymmetries of heterogeneous chiplets which makes the MILP formulation more complex.

[5]In our experiments, we use 4 pin clumps per chiplet, where each pin clump accounts for the microbumps on an edge of the chiplet [25], [26]

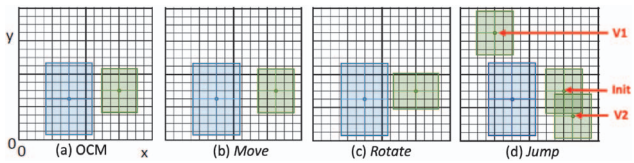


Fig. 2: (a) A logical view of the OCM data structure used by TAP-2.5D modeling two chiplets over the floorplan. (b, c, d) represent three examples of chiplet movements – V1 and V2 are two valid positions for the jump operation starting from the initial chiplet placement (Init).

2) Initial placement: Theoretically, the initial placement does not matter in an SA process as long as the process can run long enough to cover a substantial portion of the solution space. However, we want to find a satisfactory solution in a limited amount of time (as explained in Section III-C5, we calibrate our simulations to stop after 25 hours). Thus, a good initial placement is critical as it can help the SA process use the limited number of steps more efficiently, and explore the placements that are closer to the optimal choice. In our methodology, we implement the floorplanning method developed by Chen *et al.* [11] to generate an initial placement. This method uses B*-tree data structure, which is known to be the most efficient and flexible floorplan representation [30], and uses fast-SA algorithm, which efficiently searches for a solution of modern fixed-outline floorplanning problem for both area reduction and wirelength minimization. We use the compact chiplet placement solution from the B*-tree and fast-SA (we will refer to it as Compact-2.5D approach) based method as the initial placement for our methodology.

3) Neighbor placement: To find a neighbor placement, we perturb the current chiplet placement with move (Figure 2(b)), rotate (Figure 2(c)), and jump (Figure 2(d)) operations to get a new valid placement. For a rotate operation, we randomly pick a chiplet and rotate it by 90 degree. For a move operation, we randomly pick a chiplet and move it by a minimum step size ($1mm$ in our case) in up, down, left or right directions, while ensuring no chiplets overlap after the move. With only the rotate and move operations, the relative positions of the chiplets are unlikely to change. Thus, the SA process may run into the 'sliding tile puzzle' issue where a chiplet cannot move in certain directions because other chiplets block the way. To resolve this 'sliding tile puzzle' issue, we use the jump operation. With a jump operation, a randomly picked chiplet can jump to any valid empty location on the interposer. A valid neighbor placement should have no overlap between chiplets and should be completely on the interposer.

4) SA cost function: The goal of TAP-2.5D is to find an inter-chiplet routing solution while minimizing the operating temperature and the total wirelength for heterogeneous 2.5D systems with a given network connectivity. Eqn. (12) shows our SA cost function. The temperature (T) and wirelength (W) are normalized using Min-Max Scaling to alleviate the impact of imbalanced values and ranges of raw data. α and $(1-\alpha)$ are the weights of the temperature and wirelength terms, respectively. Here, α is picked by our algorithm rather than by users because we are seeking a thermally-feasible solution that also minimizes wirelength, rather than a solution with optimized wirelength but unfeasible high temperature that could immediately burn the system. So we dynamically adjust α at design time to be aware of the temperature level, as shown in Eqn. (13). At a higher temperature, our algorithm prioritizes lowering the temperature (effectively choosing an α value of greater than 0.5), which is critical to maintain safe operation. When the temperature is below 85^oC, the algorithm focuses purely on minimizing the wirelength (effectively choosing an α value of less than 0.5), as there is no point to trade off wirelength for lower temperature.

$$Cost = \alpha \times \frac{T - T_{min}}{T_{max} - T_{min}} + (1-\alpha) \times \frac{W - W_{min}}{W_{max} - W_{min}} \tag{12}$$

$$\alpha = \begin{cases} min\{0.1 + \frac{T-45}{100}, 0.9\}, & if\ T > 85^oC \\ 0, & if\ T \leq 85^oC \end{cases} \tag{13}$$

5) Acceptance probability: The decision of whether a neighbor placement is accepted or not depends on the Acceptance Probability (AP). We compute the AP using Eqn. (14), where the cost of current and neighbor placements are computed using Eqn. (12), and K is the annealing temperature, which decays from 1 to 0.01 with a factor of 0.95 (we use this value as it allows to finish each experiment within 25 hours that equals 4,500 steps). We accept the neighbor placement if AP is greater than a random number between 0 and 1. In the case that a neighbor placement is better or equal ($Cost_{neighbor} \leq Cost_{current}$), then AP value becomes greater than or equal to 1 and in that case our algorithm always accepts the neighbor placement solution. In the case that a neighbor placement is worse ($Cost_{neighbor} > Cost_{current}$), there is still a nonzero probability of accepting the worse neighbor placement to avoid getting trapped in a local minima. The worse a neighbor placement is the lower is the probability of accepting it. As the annealing temperature K decays, the solution converges because the probability of accepting a worse neighbor placement decreases.

$$AP = e^{\frac{Cost_{current} - Cost_{neighbor}}{K}} \tag{14}$$

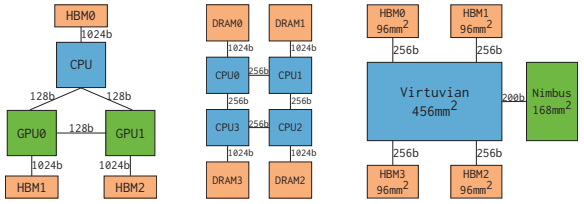
IV. Evaluation Results

In this section, we discuss the results of applying TAP-2.5D to both conceptual and existing heterogeneous 2.5D systems. The logical network topologies of the heterogeneous 2.5D systems we evaluated are shown in Fig. 3. We use publicly available data for the dimensions and power consumption of the chiplets (see Table II)[6]. Our evaluation uses $45mm \times 45mm$ interposers unless otherwise specified. This interposer size is the minimum required for the 3 systems we evaluated. Of course, for smaller systems this interposer size will be smaller. As SA is a probabilistic approach, we run the algorithm 5 times and pick the best solution. We compare our TAP-2.5D with respect to a representative state-of-the-art compact placement approach developed by Chen *et al.* [11] (B*-tree and fast-SA, called Compact-2.5D) that, as we mentioned in Section III-C2, is also used as the initial placement to speed up TAP-2.5D.

[6]In case the data is not publicly available, we apply standard technology scaling rules. Our TAP-2.5D methodology is independent of the area and power values.

TABLE II: Chiplet dimensions and powers in 2.5D examples.

	Multi-GPU System			CPU-DRAM System		Ascend 910 System		
Chiplet	CPU	GPU	HBM	CPU	DRAM	Virtuvian	Nimbus	HBM
Widths $[mm]$	12	18.2	7.75	8.25	8.75	31.4	10.5	7.75
Height $[mm]$	12	18.2	11.87	9	8.75	14.5	16	11.87
Power $[W]$	105	295	20	150	20	256	14	20



(a) (b) (c)

Fig. 3: Network topologies of (a) a conceptual Multi-GPU System, (b) CPU-DRAM System [14], and (c) Huawei Ascend 910 System [15].

A. Case Study 1: Multi-GPU System

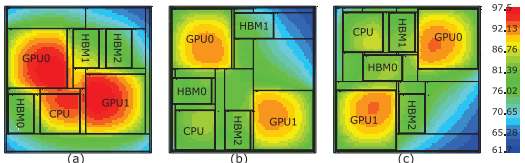
Fig. 4 shows the thermal maps of the conceptual Multi-GPU System. The placement in Fig. 4(a) is obtained by using the Compact-2.5D approach, which minimizes wirelength and area, but does not account for temperature. This system operates at 95.31^oC with a total wirelength (sum of all inter-chiplet link lengths) of $88,059mm$. Fig. 4(b) is the output from our TAP-2.5D methodology that uses a physical network with repeaterless non-pipelined inter-chiplet links. This layout has a lower peak temperature of 91.25^oC with a longer $96,906mm$ total wirelength as it pushes the high-power CPU and GPU chiplets to the corners. Fig. 4(c) is our placement solution using *gas-station* links. The temperature of the system is similarly lower (91.52^oC) but the total wirelength reduces to $51,010mm$ (vs. $88,059mm$ obtained by Compact-2.5D). This is achieved by placing the HBMs in the middle of the CPU and GPU chiplets, where the HBM chiplets provide "gas-stations" for connections between CPU and GPU chiplets.

Impact of interposer sizes: We use $45mm \times 45mm$ interposer in this case study as we can fit all chiplets in that area. When we increase the interposer size to $50mm \times 50mm$ and apply our methodology, we achieve lower temperature but longer wirelength. Compared to the $45mm \times 45mm$ interposer-based design, the $50mm \times 50mm$ interposer-based design has 2.51^oC lower temperature at 5% higher wirelength for the repeaterless non-pipelined link case, and 2.38^oC lower temperature at 17% higher wirelength for the *gas-station* link case. However, this comes at a 33% higher interposer cost.[7]

B. Case Study 2: CPU-DRAM System

Fig. 5 shows the thermal maps of the CPU-DRAM System, where (a) is the original placement [14], (b) is the placement solution using Compact-2.5D approach, (c) and (d) are our thermally-aware placement solutions using repeaterless non-pipelined inter-chiplet link and using *gas-station* links, respectively. The original placement (a) is optimal from the routing perspective (total wirelength of $67,686mm$ according to our evaluation). However, our HotSpot simulations show that the system operates at 115.94^oC, which is thermally infeasible.

[7]The increase in wirelength could lower system performance, but that can be recovered as we are reducing temperature which enables operating the system at higher voltage and frequency.



(a) (b) (c)

Fig. 4: Thermal maps of the Multi-GPU System: (a) a placement solution using Compact-2.5D approach, (b) TAP-2.5D solutions using repeaterless non-pipelined inter-chiplet links, and (c) *gas-station* links.

The placement in (b) is also relatively compact (the total wirelength is $100,864mm$), therefore, the peak temperature is 113.54^oC, which is also thermally infeasible. Our thermally-aware placement solutions in (c) and (d) successfully reduce the peak temperature to 94.89^oC and 93.89^oC, respectively. It is achieved by pushing the high-power CPU chiplets to the corners of the interposer. The total wirelengths for solutions in (c) and (d) are $216,064mm$ and $138,956mm$, respectively. *It should be noted here, we are not trading off the 2× to 3× longer wirelength (compared to the original solution (a)) for a lower temperature, it is the price we have to pay to turn a thermally-infeasible design to a thermally-feasible design.*

Impact on TDP: We complete a TDP analysis to highlight the benefit of our thermally-aware physical network design.[8] We vary the CPUs' power to determine the TDP envelopes (the maximum power of all the chiplets without violating 85^oC temperature constraint) of the original CPU-DRAM System (Fig. 5(a)) [14] and our placement solution (Fig. 5(c)). The original system shown in (a) can tolerate $400W$, and the system using our TAP-2.5D methodology shown in (c) increases the TDP to $550W$. The TDP increase is achieved by pushing the high-power chiplets away from each other to avoid heat aggregation, which needs longer inter-chiplet links. The power of inter-chiplet network is negligible from prior studies [25], [26]. Based on our evaluation using PARSEC, SPLASH2 and UHPC benchmarks, increasing the inter-chiplet link latency from 1 cycle to 2 cycles results in 5% to 18% (11% on average) performance loss, and increasing the latency from 1 cycle to 3 cycles results in 18% to 39% (25% on average) performance loss. However, the increase in TDP envelope can be leveraged to improve performance (e.g., increasing the operating frequency by 30%) without increasing cooling cost.

C. Case Study 3: Huawei Ascend 910 System

Fig. 6 shows the thermal maps of the existing Huawei Ascend 910 System [15]. The original layout of Ascend 910 System (Fig. 6(a)) already achieves minimum wirelength and is thermally-safe when running at the nominal frequency. According to our simulations, the peak temperature of Ascend 910 System is 75.48^oC, which is below the typical acceptable threshold of 85^oC, and the total wirelength is $16,426mm$. Fig. 6(b) is a placement solution that we generated using the

[8]We did not do a TDP analysis for case studies 1 and 3. For case study 1, we could vary either CPU power or GPU power, and still operate the system under the same temperature constraint. However, different combinations of CPU and GPU powers lead to different TDP envelopes. For case study 3, we achieve similar placement solution as the commercial product, and there is no change in the TDP envelope.

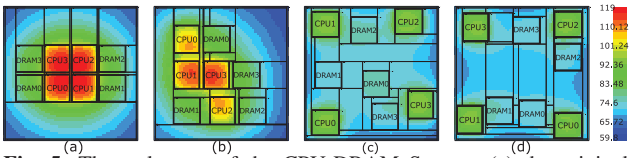


Fig. 5: Thermal maps of the CPU-DRAM System: (a) the original placement [14], (b) a placement solution using Compact-2.5D approach, (c) TAP-2.5D solutions using repeaterless non-pipelined link, and (d) using *gas-station* links.

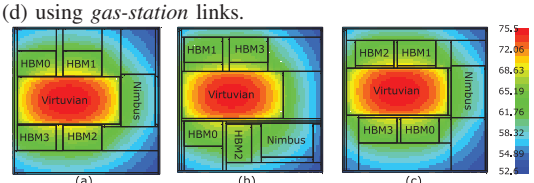


Fig. 6: Thermal maps of the Huawei Ascend 910 System [15]: (a) the original layout, (b) a placement solution using Compact-2.5D approach, and (c) TAP-2.5D solution.

chiplets and network topology of the Ascend 910 System, and Compact-2.5D approach that focuses on reducing wirelength and area. The total wirelength of the design in (b) is $23,794mm$ and the temperature is 75.13^oC. We use it as the initial placement in TAP-2.5D. Fig. 6(c) is the solution using TAP-2.5D methodology for the system (it yields the same placement solution with or without *gas-station* links). The solution has $16,597mm$ total wirelength and 75.47^oC temperature. Effectively, our open-source placement solution is comparable to the actual solution of the commercial chip.

D. Discussion on Scalability

The case studies we have shown are small-to-medium sized 2.5D system examples with up to 8 chiplets. Our methodology also supports heterogeneous 2.5D systems with a large number of chiplets, but requires longer simulation time. The bottlenecks of our approach are the thermal analysis (each HotSpot simulation takes 23 seconds on average) and routing optimization (each MILP operation takes 5 seconds on average). The thermal evaluation time is independent of the chiplet count, as we use a fixed grid size (64×64) for the systems. The time spent on routing optimization scales with $O(|C|^2 \cdot |P|^2 \cdot |N|)$, where $|C|$, $|P|$, $|N|$ are the number of chiplets, the number of pin clumps per chiplet, and the number of inter-chiplet channels, respectively. As part of our future work, we will explore the use of machine learning techniques to accelerate the thermal analysis and routing optimization.

V. Conclusions

In this paper, we propose TAP-2.5D, an inter-chiplet physical network design methodology for heterogeneous 2.5D systems. The goal of our methodology is to find the physical design solution for an inter-chiplet network by jointly minimizing the operating temperature of the overall system and total inter-chiplet network wirelength. Our methodology strategically inserts spacing between chiplets to improve heat dissipation, and thus increases the thermal design power of the overall system. We develop a simulated annealing based approach, which searches for a thermally-aware chiplet placement and optimizes the routing of inter-chiplet wires for heterogeneous 2.5D systems. We demonstrate the usage of TAP-2.5D by applying it to three heterogeneous 2.5D systems.

Acknowledgment

This work was supported in part by NSF CCF-1716352 and NSF CNS-1525474 grants.

References

[1] "Heterogeneous integration roadmap 2019 edition," https://eps.ieee.org/technology/heterogeneous-integration-roadmap/2019-edition.html.
[2] "FPGA VC707 evaluation kit," Virtex-7, Xilinx.
[3] R. Radojcic, *More-than-Moore 2.5D and 3D SiP Integration*. Springer, 2017.
[4] J. Macri, "AMD's next generation GPU and high bandwidth memory architecture: FURY," in *Proc. HCS*, 2015, pp. 1–26.
[5] "Nvidia: NVIDIA Tesla P100," https://images.nvidia.com/content/pdf/tesla/whitepaper/pascal-architecture-whitepaper.pdf.
[6] "Intel introduces Foveros: 3D die stacking for more than just memory," https://arstechnica.com/gadgets/2018/12/intel-introduces-foveros-3d-die-stacking-for-more-than-just-memory/, 2018.
[7] "Hot Chips 2017: Intel Deep Dives Into EMIB," https://www.tomshardware.com/news/intel-emib-interconnect-fpga-chiplet,35316.html.
[8] S. Ramalingam, "HBM package integration: Technology trends, challenges and applications," in *Proc. HCS*, 2016, pp. 1–17.
[9] R. Mahajan *et al.*, "Embedded multi-die interconnect bridge (EMIB)–a high density, high bandwidth packaging interconnect," in *Proc. ECTC*, 2016, pp. 557–565.
[10] G. Parès, "3D interposer for silicon photonics," *LETI Innovations Days*, 2013.
[11] T. Chen *et al.*, "Modern floorplanning based on b/sup*/-tree and fast simulated annealing," *IEEE TCAD*, vol. 25, no. 4, pp. 637–650, 2006.
[12] A. Coskun *et al.*, "Energy-efficient variable-flow liquid cooling in 3D stacked architectures," in *Proc. DATE*, 2010, pp. 111–116.
[13] F. Eris *et al.*, "Leveraging thermally-aware chiplet organization in 2.5D systems to reclaim dark silicon," in *Proc. DATE*, 2018, pp. 1441–1446.
[14] A. Kannan *et al.*, "Enabling interposer-based disintegration of multi-core processors," in *Proc. MICRO*, 2015, pp. 546–558.
[15] "Huawei ascend 910 provides a nvidia ai training alternative," https://www.servethehome.com/huawei-ascend-910-provides-a-nvidia-ai-training-alternative/.
[16] J. Kim *et al.*, "Architecture, chip, and package co-design flow for 2.5D IC design enabling heterogeneous IP reuse," in *Proc. DAC*, 2019, pp. 1–6.
[17] T. Vijayaraghavan *et al.*, "Design and analysis of an APU for exascale computing," in *Proc. HPCA*, 2017, pp. 85–96.
[18] M. Ebrahimi *et al.*, "NoD: Network-on-Die as a standalone NoC for heterogeneous many-core systems in 2.5D ICs," in *Proc. CADS*, 2017, pp. 1–6.
[19] J. Yin *et al.*, "Modular routing design for chiplet-based systems," in *Proc. ISCA*, 2018, pp. 726–738.
[20] H. Murata *et al.*, "VLSI module placement based on rectangle-packing by the sequence-pair," *IEEE TCAD*, vol. 15, no. 12, pp. 1518–1524, 1996.
[21] J. Lin *et al.*, "TCG: A transitive closure graph-based representation for non-slicing floorplans," in *Proc. DAC*, 2001, pp. 764–769.
[22] P. Guo *et al.*, "An O-tree representation of non-slicing floorplan and its applications," in *Proc. DAC*, 1999, pp. 268–273.
[23] M. Healy *et al.*, "Multiobjective microarchitectural floorplanning for 2-D and 3-D ICs," *IEEE TCAD*, vol. 26, no. 1, pp. 38–52, 2006.
[24] J. Cong *et al.*, "A thermal-driven floorplanning algorithm for 3D ICs," in *Proc. ICCAD*. IEEE, 2004, pp. 306–313.
[25] A. Coskun *et al.*, "A cross-layer methodology for design and optimization of networks in 2.5D systems," in *Proc. ICCAD*, 2018, pp. 101–108.
[26] ——, "Cross-layer co-optimization of network design and chiplet placement in 2.5 d systems," *IEEE TCAD*, 2020.
[27] J. Meng *et al.*, "Optimizing energy efficiency of 3-D multicore systems with stacked DRAM under power and thermal constraints," in *Proc. DAC*, 2012, pp. 648–655.
[28] R. Chaware *et al.*, "Assembly and reliability challenges in 3D integration of 28nm FPGA die on a large high density 65nm passive interposer," in *Proc. ECTC*, 2012, pp. 279–283.
[29] J. Charbonnier *et al.*, "High density 3D silicon interposer technology development and electrical characterization for high end applications," in *Proc. ESTC*, 2012, pp. 1–7.
[30] N. Laskar *et al.*, "A survey on vlsi floorplanning: its representation and modern approaches of optimization," in *Proc. ICIIECS*, 2015, pp. 1–9.