# Exploiting Die-to-Die Thermal Coupling in 3-D IC Placement

Krit Athikulwongse, *Member, IEEE*, Mongkol Ekpanyapong, *Senior Member, IEEE*, and Sung Kyu Lim, *Senior Member, IEEE*

***Abstract*— In this paper, we propose two methods used in 3-D IC placement that efficiently exploit the die-to-die thermal coupling in the stack. First, through-silicon vias (TSVs) are spread on each die to reduce the local power density and vertically aligned across dies simultaneously to increase thermal conductivity to the heatsink. Second, we move high-power logic cells to the location that has higher conductivity to the heatsink while moving TSVs in the upper dies so that high-power cells are vertically overlapping below the TSVs. These methods are employed in a force-directed 3-D placement successfully and outperform several state-of-the-art placers published in recent literature. We obtain 3-D placement results with shorter routed wirelength at similar temperature. We also obtain 3-D placement results with lower temperatures at similar routed wirelengths.**

***Index Terms*— 3-D IC, placement, temperature, through-silicon vias (TSV).**

## I. Introduction

INCREASING functionality while miniaturizing the footprint of integrated circuits (ICs) is today's trend of electronic industry. Moving to a smaller technology node is a traditional approach toward that goal; however, investing in new production lines needs to be economically justified. 3-D stacking of thinned dies provides feasibility to keep the trend while staying at the current technology node. Polymer adhesive is a popular material used to bond thinned dies together [1]. Interleaving layers of thinned dies and polymer adhesive are, therefore, commonly found in 3-D ICs.

Stacking thinned dies in 3-D ICs results in increasing power density, thus rising temperature, which leads to other reliability problems, such as electromigration and negative-bias-temperature instability. Because of low-thermal conductivity, polymer adhesive exacerbates the problem. Moreover, if the thinned dies are silicon on insulator, an extremely high temperature can be expected. Heat must be removed from the die quickly; otherwise, reliability problems may arise.

A few recent works on temperature-aware placement for 3-D ICs have been published. In [2], Chen *et al.* provide an overview of thermal modeling and simulation for 2-D and 3-D placement. Cong *et al.* [3] proposed an algorithm to transform 2-D placement to 3-D placement. Yan *et al.* [4] proposed a quadratic placement for unified cell distribution and thermal dissipation for 3-D ICs. In [5], a force-directed approach was proposed for 3-D thermal placement; however, it did not include through-silicon vias (TSVs), which are commonly found in 3-D ICs. In [6], a partitioning-based approach was proposed for 3-D thermal placement. The work considered the impact of parasitic resistance and capacitance of signal TSVs on power, but failed to include thermal properties of TSVs. Failing again to acknowledge TSV area, it also reported unreasonably large numbers of TSVs even for small circuits. The work in [7] considered TSV thermal properties; however, it assumed that adhesive is an ideal insulator. In reality, heat can still flow through (silicon and) adhesive because of its thinness. Based on the assumption, the work balanced only the number of TSVs in a bin to heat dissipated from cells in the same bin and bins vertically below. The challenge for 3-D Design, especially physical design, has been addressed in [8]. All of this papers failed to address the important of TSV dimension and its thermal property (e.g., thermal conductivity and power white space).

The contributions of this paper are as follows.

1) We propose two effective heuristics, namely TSV spread and alignment method (TSA) and thermal coupling-aware placement (CA), that exploit the die-to-die thermal coupling in 3-D ICs in force-directed temperature-aware placement. We present new forces, and discuss how to manage them to obtain high-quality placements.
2) We present our framework to evaluate the impact of TSVs on temperature of 3-D ICs. The main components of the framework are power analysis and GDSII-level thermal analysis for 3-D ICs.

We perform extensive experiments to show the trade-off among wirelength, delay, power, and temperature results obtained from GDSII layouts. Our placers outperform several state-of-the-art placers published in recent literature [5]–[7], [9], [10]. We obtain 3-D placement results with shorter routed wirelength at similar temperature. We also obtain 3-D placement results with lower temperature at similar routed wirelength.

The rest of this paper is organized as follows. We explain our motivation in Section II. The two effective heuristics proposed in this paper are described in Section III. We present our framework to evaluate the impact of TSVs on temperature

Manuscript received February 11, 2013; revised August 13, 2013; accepted September 29, 2013. Date of publication October 23, 2013; date of current version September 23, 2014. This work was supported by Semiconductor Research Corporation under Grant Task ID 1836.075.

K. Athikulwongse is with the National Electronics and Computer Technology Center, Khlong Luang 12120, Thailand (e-mail: krit@gatech.edu).

M. Ekpanyapong is with Microelectronics and Embedded Systems Department, Asian Institute of Technology, Thailand (e-mail: mongkol@ait.ac.th).

S. K. Lim is with the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: limsk@ece.gatech.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TVLSI.2013.2285593

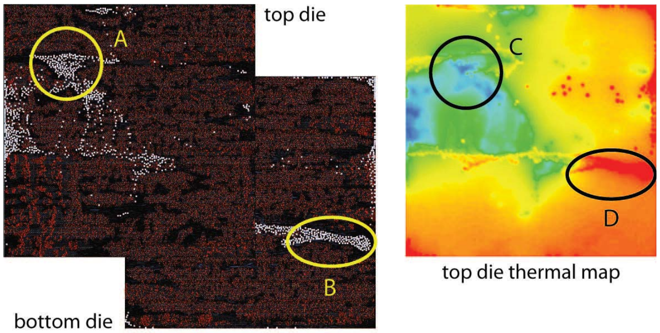

Fig. 1. Die-to-die heat coupling from TSVs. TSVs are shown in white. The top die is closer to heatsink. The cold spot C is caused by the TSVs in spot A on the same die. The hot spot D is caused by the TSVs in spot B from the bottom die.

of 3-D ICs in Section IV. Experimental results are shown in Section V, and we conclude in Section VI.

## II. Motivation

Because of their occupied area and high-thermal conductivity of copper, widely used fill material, and TSVs have significant impact on temperature. In a 3-D IC layout, logic gates cannot overlap with TSVs. Area occupied by TSVs becomes "power whitespace" because no power is consumed and thus no heat is generated. In addition, TSVs conduct majority of heat through polymer adhesive between dies toward the heatsink as shown in Fig. 1. In the figure, the hotspot D on the top metal layer of the top die is caused by the TSVs in spot B from the bottom die. Heat flows through TSVs so intensely that its effect still remains on the top die. Thus, the temperature distribution of the top die results from the combination of power profile of the top die and heat flowing from the bottom die through TSVs. Our TSA presented in this paper, exploits these thermal properties of TSVs by distributing TSVs evenly to reduce power density in local power hotspots and vertically aligning TSVs of adjacent dies to establish direct paths to the heatsink.

## III. Global Placement Algorithms

In this section, we describe our two 3-D temperature-aware global placement algorithms that are based on the force-directed methodology [11]. We extend this placer in two ways to perform thermal optimization in 3-D ICs. In the first algorithm, we laterally spread TSVs in each die to form even thermal conductivity while perturbing TSV position to increase vertical overlap among TSVs across the dies in a 3-D stack. In the second algorithm, the logic cells on each die are positioned by using thermal conductivity-based force while TSVs are positioned by using power density-based force.[1]

### A. Design Flow

Fig. 2 shows the overall flow of our placement, where the position of cells and TSVs is determined simultaneously.

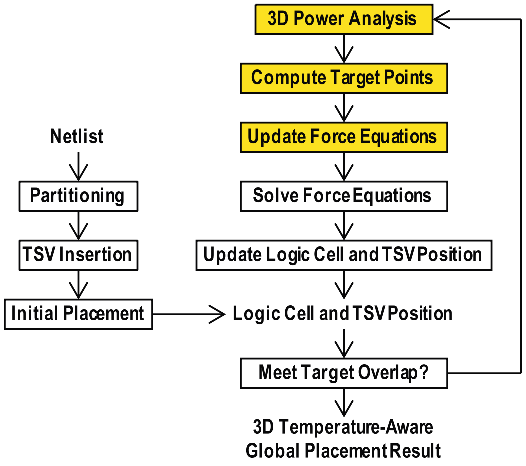

Fig. 2. Design flow for our 3-D IC global placement.

Given a netlist, we partition cells into dies if the partition is not also given. Then, we insert the minimum number of TSVs required to connect cells on different dies. Once this die partitioning is fixed, we do not move cells across dies during placement. The reason is that changing cell partition results in change in the number of TSVs, and this change causes the complexity of problem to become unmanageable. Next, we minimize wirelength to obtain initial placement, which may contain high overlap among cells and TSVs. The initial placement is obtained by minimizing the quadratic wirelength over a few iterations as in [11]. The quadratic wirelength is computed from the 3-D netlist [10], that includes both cells and TSVs. TSVs are treated as cells because they occupy placement area. This initial placement has a minimal wirelength, however, it contains significant overlap between cells, between TSVs, and between cells and TSVs. In the main loop to resolve the overlap, we use TSV density and TSV position to compute target point for TSVs in the first algorithm. In the second algorithm, we periodically perform 3-D power analysis (explained in Section IV-A) based on current cell and TSV position. Then, we use the cell power, TSV density, and average thermal conductivity of bulk silicon obtained from the simulation results in Section II to compute target points for cells and TSVs to move toward. After updating force equations and solving them, we update the position of cells and TSVs. This loop continues until the overlap is sufficiently reduced.[2]

### B. Overview of Kraftwerk Framework

In a quadratic placement [11], quadratic wirelength $\Gamma_x$ and $\Gamma_y$ along $x$- and $y$-axis are separately minimized to obtain the placement result. Treated $\Gamma_x$ as spring energy, its derivative can be regarded as net force $\mathbf{f}_x^{\text{net}}$. By setting $\mathbf{f}_x^{\text{net}}$ to zero, the minimum $\Gamma_x$ and the corresponding placement are found, however, cells may overlap in few small areas. Hold force $\mathbf{f}_x^{\text{hold}}$

[1] We attempted combining these two methods, but the results were not consistent.

[2] Timing is not explicitly optimized in our flow. But, wirelength minimization indirectly affects path delays. We do report longest path delay in the experimental result section, not as the primary objective but to show the impact from thermal optimization. One quick way to optimize timing is to assign timing-based weights to each net and minimize weighted wirelength, but this may cause temperature increase.

prevents $\mathbf{f}_{\text{x}}^{\text{net}}$ from pulling cells back to the initial placement. In addition, density-based force $\mathbf{f}_{\text{x}}^{\text{den}}$ reduces the overlap by spreading cells in high-density region.

To extend [11] for 3-D ICs, cells are not moved across dies during placement in [10] because they are already assigned into dies by the partitioner. In addition, $\mathbf{f}_{\text{x}}^{\text{den}}$ is computed die-by-die based on the placement density $D_d$ of each die $d$, which is defined as

$$D_d(x, y) = D_d^{\text{cell}}(x, y) - D_d^{\text{die}}(x, y) \qquad (1)$$

where $D_d^{\text{cell}}$ is the cell density on die $d$, and $D_d^{\text{die}}$ is the die-capacity scaled to match the total cell area on the die. Then, the placement potential $\Phi_d$ is computed by solving Poisson's equation

$$\Delta \Phi_d(x, y) = -D_d(x, y). \qquad (2)$$

The target point $\mathring{x}_i^{\text{d}}$ to connect density-based spring of cell $i$ is computed by

$$\mathring{x}_i^{\text{d}} = x_i' - \frac{\partial}{\partial x}\Phi_d(x, y)\Big|_{(x_i', y_i')} \qquad (3)$$

where $x_i'$ is the x-position of cell $i$ on die $d$ from the last iteration. The negative gradient of $\Phi_d$ indicates in which direction and how fast the cell at that position should move. We model $\mathbf{f}_{\text{x}}^{\text{den}}$ by connecting cell $i$ to its target point $\mathring{x}_i^{\text{d}}$ with a spring of spring constant $\mathring{w}_i^{\text{d}}$. Therefore, for cell $i$, density-based force $f_{\text{x},i}^{\text{den}} = \mathring{w}_i^{\text{d}}(x_i - \mathring{x}_i^{\text{d}})$, where $x_i$ is the x-position of cell $i$ being placed. Finally, $\mathbf{f}_{\text{x}}^{\text{den}}$ is defined for 3-D ICs by

$$\mathbf{f}_{\text{x}}^{\text{den}} = \mathring{\mathbf{C}}_{\text{x}}^{\text{d}}(\mathbf{x} - \mathring{\mathbf{x}}^{\text{d}}) \qquad (4)$$

where $\mathring{\mathbf{C}}_{\text{x}}^{\text{d}}$ is a diagonal matrix of $\mathring{w}_i^{\text{d}}$, $\mathbf{x} = [x_1, \ldots, x_N]^{\text{T}}$ is a vector representing the x-position of $N$ cells being placed, and $\mathring{\mathbf{x}}^{\text{d}} = [\mathring{x}_1^{\text{d}}, \ldots, \mathring{x}_N^{\text{d}}]^{\text{T}}$ is a vector representing the target x-position of the cells. We set $\mathring{w}_i^{\text{d}}$ proportional to the area of each cell so that big cells are moved by strong spring. Lastly, for each placement iteration, the placement result can be obtained by setting total force $\mathbf{f}_{\text{x}}$ to zero, and solve

$$\mathbf{f}_{\text{x}} = \mathbf{f}_{\text{x}}^{\text{net}} + \mathbf{f}_{\text{x}}^{\text{hold}} + \mathbf{f}_{\text{x}}^{\text{den}} = \mathbf{0}. \qquad (5)$$

For each iteration, if no overlap remains in the placement, $\mathbf{f}_{\text{x}}^{\text{den}}$ becomes $\mathbf{0}$. Then, $\mathbf{f}_{\text{x}}^{\text{hold}}$ balances out $\mathbf{f}_{\text{x}}^{\text{net}}$, and the position of all cells does not change. However, if some overlap remains in the placement, $\mathbf{f}_{\text{x}}^{\text{den}}$ perturbs the balance of $\mathbf{f}_{\text{x}}^{\text{hold}}$ and $\mathbf{f}_{\text{x}}^{\text{net}}$. The position of all the cells changes to balance out the perturbation, resulting in the new placement with decreasing overlap.

Equation (5) is solved once in each iteration during the global placement stage. The dimension of the equation is the number of cells and TSVs (recall that TSVs are treated as cells because they occupy placement area). The resulting $\mathbf{x}$ contains the new x-position of all the cells and TSVs on all the dies. By solving (5), the new x-position of all the cells and TSVs on all the dies is obtained simultaneously [10], [11].

### *C. TSV Spread and Alignment*

In this algorithm, we exploit one of thermal properties of TSVs to help alleviate thermal problems as shown in Fig. 3(a). TSVs occupy placement area, but do not dissipate power.[3] The existence of TSVs among cells with high-power dissipation reduces local dissipated power density, which in-turn helps reduce local temperature. Therefore, for gate-level placement of cells with similar power density, spreading TSVs evenly on each die should help reduce intradie thermal variation in 3-D ICs.[4] We propose this algorithm because it is simple yet effective. It can be viewed as a method to mimic uniform TSV position. Instead of moving TSVs based on the placement density computed from both TSV and cell area, we move TSVs based on TSV density only. In other words, we compute $D_d^{\text{cell}}$ in (1) from TSV area only, and scale $D_d^{\text{die}}$ to match the total TSV area on the die.

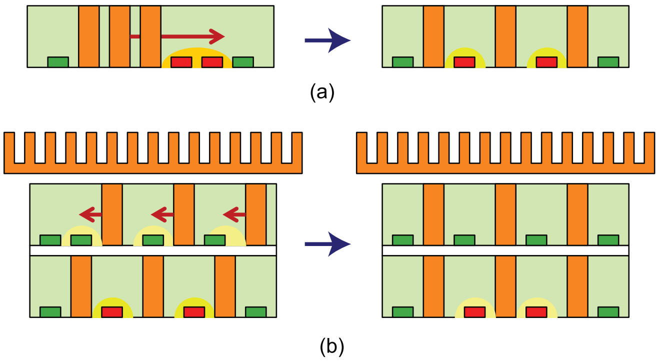

Fig. 3. (a) TSV spread and (b) TSV align forces.

In addition to TSV spread, we exploit another thermal property of TSVs to help alleviate thermal problems as shown in Fig. 3(b). TSVs conduct majority of heat through polymer adhesive between dies, causing local hot spots on the adjacent die between the TSVs and heatsink. Therefore, aligning TSVs on each die to TSVs on the adjacent die should help prevent this kind of hot spots, and direct the heat toward the heatsink quickly, resulting in overall temperature decrease. To align TSVs during global placement, we introduce an additional force for TSVs, alignment force denoted $\mathbf{f}_{\text{x}}^{\text{align}}$, into (5). This force can be represented by alignment springs connected to TSVs, and defined as

$$\mathbf{f}_{\text{x}}^{\text{align}} = \mathring{\mathbf{C}}_{\text{x}}^{\text{a}}(\mathbf{x} - \mathring{\mathbf{x}}^{\text{a}}) \qquad (6)$$

where vector $\mathring{\mathbf{x}}^{\text{a}}$ represents the x-position of target points to connect alignment springs to TSVs, and diagonal matrix $\mathring{\mathbf{C}}_{\text{x}}^{\text{a}}$ collects spring constants $\mathring{w}_{\text{x},i}^{\text{a}}$ of the alignment spring connected to TSV $i$.

We align each TSV on a die to another TSV on the adjacent die below. The forces that move TSVs on a die are $\mathbf{f}_{\text{x}}^{\text{den}}$ and $\mathbf{f}_{\text{x}}^{\text{align}}$. The placement density of the same die as the TSVs determines $\mathbf{f}_{\text{x}}^{\text{den}}$ while TSV position on the adjacent die below determines $\mathbf{f}_{\text{x}}^{\text{align}}$. The two forces compete with each other. In the early placement iterations, $\mathbf{f}_{\text{x}}^{\text{den}}$ dominates because of highly uneven placement density. In the late placement

[3] The percentage of total TSV landing pad area depends on die partitioning, TSV dimension, and keep-out-zone size. In our study [12], the ratio ranges from 1% to 34%.

[4] Note that this strategy may not be applicable for mixed-size placement, where there exists a significant size unbalance between large macros and gates.

iterations, $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ becomes weak because of leveled placement density, allowing $\mathbf{f}_{\mathrm{x}}^{\mathrm{align}}$ to impact TSV position. If a TSV on a die overlaps (even very slightly) the nearest TSV on the adjacent die below, we set the target point of the alignment spring of the TSV to the position of the nearest TSV. Therefore, the TSV is moved toward the nearest TSV in the next iteration of global placement, improving the alignment of both TSVs. In the case that the adjacent die below has fewer TSVs, it is possible that some TSVs on a die cannot find another TSV on the adjacent die below to align to. We do not apply $\mathbf{f}_{\mathrm{x}}^{\mathrm{align}}$ to these TSVs in such case. In other words, we apply alignment force to TSV $i$ only when its closest TSV $j$ on the adjacent die farther from the heatsink is within a certain range so that we do not excessively increase wirelength. The range is set to the size of TSV because of the high probability of aligning the TSVs in few iterations. We balance $\mathbf{f}_{\mathrm{x}}^{\mathrm{align}}$ against other forces by setting $\mathring{w}_{\mathrm{x},i}^{\mathrm{a}}$ to density-based spring constant $\mathring{w}_{\mathrm{x},i}^{\mathrm{d}}$ of $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ and setting alignment target point $\mathring{x}_i^{\mathrm{a}}$ to $x'_j$, the x-position from last iteration of TSV $j$ (on the adjacent die farther from heatsink) closest to TSV $i$. This method naturally balances $\mathbf{f}_{\mathrm{x}}^{\mathrm{align}}$ against $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$.

In other words, we set the weight of alignment spring to the same weight as density-based spring, and balance $\mathbf{f}_{\mathrm{x}}^{\mathrm{align}}$ against $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ by setting the target point of alignment spring and density-based spring differently. The target point of density-based spring is determined by solving Poisson's equation and taking the gradient of electrical potential, whereas the target point of alignment spring is determined by the position of the nearest TSV on the adjacent die below (if they overlap even very slightly). The intuition is that because of the high-cell overlap in the early placement iterations, the target point $\mathring{x}_i^{\mathrm{d}}$ is farther away from TSV $i$ than the alignment target point $\mathring{x}_i^{\mathrm{a}}$. Thus, $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ dominates. When cells are evenly distributed in the late iterations of placement, $\mathring{x}_i^{\mathrm{d}}$ is closer to TSV $i$. Then, $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ becomes weaker, and $\mathbf{f}_{\mathrm{x}}^{\mathrm{align}}$ affects the TSV position more.

### D. Thermal Coupling-Aware Placement

In this algorithm, we consider the die-to-die thermal coupling during placement. Recall that the steady-state temperature of a point $\mathbf{p} = (x, y, z)$ inside a 3-D structure can be obtained by solving the heat equation $\nabla \cdot (k(\mathbf{p})\nabla T(\mathbf{p})) + S_{\mathrm{h}}(\mathbf{p}) = 0$, where $k$ is thermal conductivity, $T$ is temperature, and $S_{\mathrm{h}}$ is volumetric heat source. In other words, the temperature of a region depends on the power density and thermal conductivity of the region. Therefore, the basic approach is to introduce two new forces, the first that moves cells and the second that moves TSVs, both in an attempt to place high-power cells closer to the TSV-to-heatsink path. Since the heat dissipated by a cell must flow toward heatsink, we place cells based on their power density and the effective thermal conductivity computed using the same die and the dies above. In addition, since TSV conducts heat without raising temperature too much, we place TSVs based on the total power density of the same die and the dies below.

Our basic approach is that the area with high-power density and low-thermal conductivity leads to high temperature. Thus, the temperature at a certain position depends on the difference (or imbalance) between power density and thermal conductivity. The force that moves cells (TSVs) on a die also changes the power density (thermal conductivity) distribution of the die. Our goal is to use these forces to balance the power density and the thermal conductivity at each position on the die. The force in an area with high difference should be stronger than the force in an area with low difference. The strength of a spring force depends on the distance to the connection point, so we set the strength based on this difference. Based on this concept, we first build a map of the difference, and smooth the map in an iterative fashion. Table I shows the notations used in this section.

TABLE I

NOTATIONS USED FOR THERMAL COUPLING-AWARE PLACEMENT

| | |
|---|---|
| $P_d^{\mathrm{cell}}$ | cell power density of each die $d$ |
| $K_d^{\mathrm{sink}}$ | effective thermal conductivity from die $d$ to heatsink |
| $K_d^{\mathrm{die}}$ | thermal conductivity across the opposite sides of die $d$ |
| $p_i$ | power of cell $i$ |
| $N_d^{\mathrm{TSV}}$ | total number of TSVs on die $d$ |
| $N_{\mathrm{die}}$ | number of dies |
| $B_d^{\mathrm{cond}}$ | balance factor for the thermal conductivity-based force on die $d$ |
| $s_d^{\mathrm{cond}}$ | scaling factor to match the effective thermal conductance to heatsink to cell power on die $d$ |
| $B_d^{\mathrm{pow}}$ | balance factor for the power density-based force on die $d$ |
| $s_d^{\mathrm{pow}}$ | scaling factor to match the cell power of die $d$ and below to the thermal conductance of die $d$ |
| $s_d^{\mathrm{PD}}$ | scaling factor to normalize the cell power to the cell area on die $d$ |
| $s_d^{\mathrm{KD}}$ | scaling factor to normalize the thermal conductance of die $d$ to the cell area on die $d$ |
| $\alpha$ | weighting constant for thermal coupling forces |

*1) For Cell Movement:* We introduce the thermal conductivity-based force $\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}}$ as illustrated in Fig. 4(a). It moves high-power cells toward the position with high-thermal conductivity to heatsink, and is defined as

$$\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}} = \mathring{\mathbf{C}}_{\mathrm{x}}^{\mathrm{c}}(\mathbf{x} - \mathring{\mathbf{x}}^{\mathrm{c}}) \tag{7}$$

where the vector $\mathring{\mathbf{x}}^{\mathrm{c}}$ represents the x-position of target points to connect thermal conductivity-based springs to cells, and the diagonal matrix $\mathring{\mathbf{C}}_{\mathrm{x}}^{\mathrm{c}}$ contains spring constants $\mathring{w}_{\mathrm{x},i}^{\mathrm{c}}$ of the spring connected to cell $i$.

We compute $\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}}$ die-by-die by balancing the cell power density $P_d^{\mathrm{cell}}$ of each die $d$ against its effective thermal conductivity to heatsink, denoted $K_d^{\mathrm{sink}}$. Under the demand-supply system of the force-directed framework in [11], $P_d^{\mathrm{cell}}$ and $K_d^{\mathrm{sink}}$ represent the demand and supply to remove the heat from die $d$ in the 3-D stack. We regard $P_d^{\mathrm{cell}}$ as demand because it represents the heat that needs to be removed, and regard $K_d^{\mathrm{sink}}$ as supply because it represents the thermal conductivity to heatsink, which is the way to remove such heat. We define the thermal conductivity-based balance factor $B_d^{\mathrm{cond}}$ for die $d$ as (see Fig. 5)

$$B_d^{\mathrm{cond}}(x, y) = P_d^{\mathrm{cell}}(x, y) - s_d^{\mathrm{cond}} \cdot K_d^{\mathrm{sink}}(x, y) \tag{8}$$

where $s_d^{\mathrm{cond}}$ is a scaling factor to match $K_d^{\mathrm{sink}}$ to $P_d^{\mathrm{cell}}$ across the die. We use $s_d^{\mathrm{cond}}$ to balance the total supply $\left(K_d^{\mathrm{sink}}\right)$ and

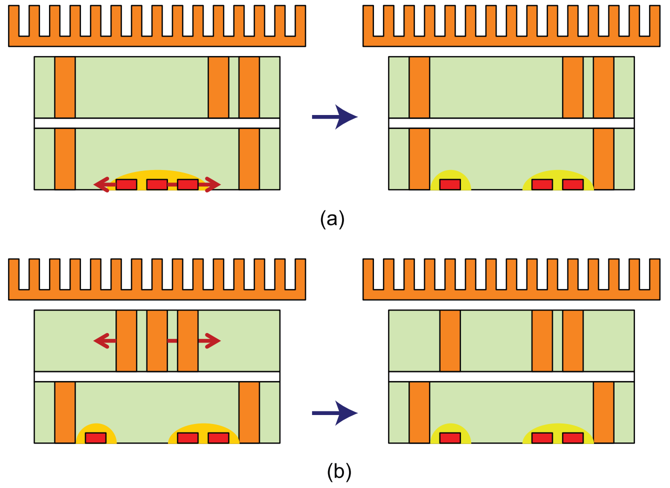

Fig. 4. Thermal conductivity-based versus power density-based forces. (a) Thermal conductivity-based force for cells and (b) power density-based force for TSVs.

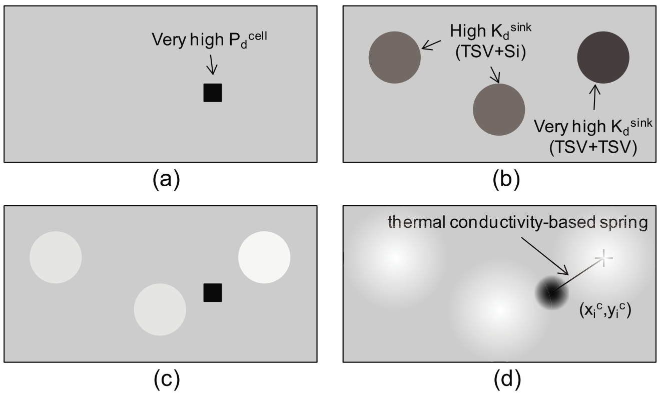
Very high $P_d^{cell}$
High $K_d^{sink}$ (TSV+Si)
Very high $K_d^{sink}$ (TSV+TSV)
thermal conductivity-based spring
$(x_i^c,y_i^c)$
(a)
(b)
(c)
(d)

Fig. 5. Illustration of $B_d^{\mathrm{cond}}$. (a) $P_d^{\mathrm{cell}}$, (b) $s_d^{\mathrm{cond}} \cdot K_d^{\mathrm{sink}}$, (c) $B_d^{\mathrm{cond}}$, and (d) potential for $B_d^{\mathrm{cond}}$ after solving Poisson's equation.

the total demand $\left(P_d^{\mathrm{cell}}\right)$, and compute it by

$$s_d^{\mathrm{cond}} = \frac{\int\int P_d^{\mathrm{cell}}(x,y)\,dx\,dy}{\int\int K_d^{\mathrm{sink}}(x,y)\,dx\,dy}. \tag{9}$$

Here, $K_d^{\mathrm{sink}}$ is computed as

$$K_d^{\mathrm{sink}}(x,y) = \frac{1}{\sum_{j=d}^{N_{\mathrm{die}}} \frac{1}{K_j^{\mathrm{die}}(x,y)}} \tag{10}$$

where $K_j^{\mathrm{die}}$ is the thermal conductivity of die $j$, and die $N_{\mathrm{die}}$ is the die closest to the heatsink (see Fig. 6). Here, $K_{N_{\mathrm{die}}}^{\mathrm{die}}$ includes the thermal conductivity of the thick substrate and heatsink, and $K_j^{\mathrm{die}}$ is computed based on the TSV density at each position on the die and the average thermal conductivity of bulk silicon with and without TSVs, obtained from the simulation results in Section II. We do not consider lateral thermal conductivity in (10) because of complexity. To consider it accurately, the lateral conductivity from each position to the entire placement area (and the vertical conductivity of all the dies above the surrounding area) must be included in the equation. We may reduce the scope to the area nearby each position; however, the scope itself may not be clearly defined because lateral conductivity also changes with layout. In addition, the impact of vertical conductivity of nearby area of each position is already indirectly considered by smoothening $B_d^{\mathrm{cond}}$.

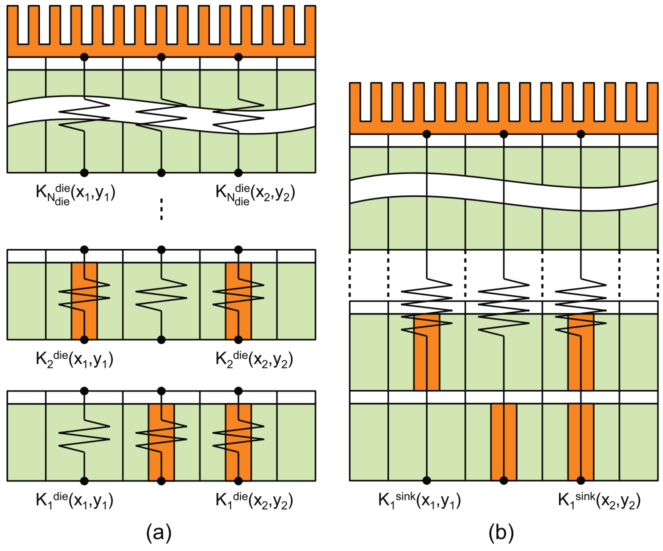

Fig. 6. Computation of $K_d^{\mathrm{sink}}$. (a) $K_j^{\mathrm{die}}$ and (b) $K_1^{\mathrm{sink}}$.

The potential $\Phi_d^{\mathrm{cond}}$ for $B_d^{\mathrm{cond}}$ is computed by solving Poisson's equation

$$\Delta\Phi_d^{\mathrm{cond}}(x,y) = -B_d^{\mathrm{cond}}(x,y). \tag{11}$$

The target point $\mathring{x}_i^{\mathrm{c}}$ of cell $i$ is computed by

$$\mathring{x}_i^{\mathrm{c}} = x_i' - \frac{\partial}{\partial x}\Phi_d^{\mathrm{cond}}(x,y)\Big|_{(x_i',y_i')} \tag{12}$$

where $x_i'$ is the x-position of cell $i$ on die $d$ from the last iteration. We set spring constant $\mathring{w}_{\mathrm{x},i}^{\mathrm{c}}$ for cell $i$ based on cell power and the total cell power by

$$\mathring{w}_{\mathrm{x},i}^{\mathrm{c}} = p_i \Big/ \sum_{\forall j} p_j \tag{13}$$

where $p_i$ is the power of cell $i$, and $j$ is a cell on die $d$. Therefore, a high-power cell is connected to a strong thermal conductivity-based spring.

*2) For TSV Movement:* We introduce power density-based force $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$ as illustrated in Fig. 4(b). It moves TSVs toward the position with high-cell power density on the same die and the dies below. We define $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$ as

$$\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}} = \mathring{\mathbf{C}}_{\mathrm{x}}^{\mathrm{p}}(\mathbf{x} - \mathring{\mathbf{x}}^{\mathrm{p}}) \tag{14}$$

where the vector $\mathring{\mathbf{x}}^{\mathrm{p}}$ represents the x-position of target points to connect power density-based springs to TSVs, and the diagonal matrix $\mathring{\mathbf{C}}_{\mathrm{x}}^{\mathrm{p}}$ contains spring constants $\mathring{w}_{\mathrm{x},i}^{\mathrm{p}}$ of the spring connected to TSV $i$.

We compute $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$ die-by-die by balancing the thermal conductivity $K_d^{\mathrm{die}}$ of each die $d$ against the total power density $\sum P_j^{\mathrm{cell}}$ that flows through the die toward heatsink. Under the demand-supply system of the force-directed framework in [11], $K_d^{\mathrm{die}}$ and $\sum P_j^{\mathrm{cell}}$ represent the demand and supply to conduct heat from the same die and dies below to heatsink. We define the power density-based balance factor $B_d^{\mathrm{pow}}$ for die $d$ as

$$B_d^{\mathrm{pow}}(x,y) = K_d^{\mathrm{die}}(x,y) - s_d^{\mathrm{pow}} \cdot \sum_{j=1}^{d} P_j^{\mathrm{cell}}(x,y) \tag{15}$$

where $s_d^{\mathrm{pow}}$ is a scaling factor to match $\sum P_j^{\mathrm{cell}}$ to $K_d^{\mathrm{die}}$ across the die. We use $s_d^{\mathrm{pow}}$ to balance the total supply $(\sum P_j^{\mathrm{cell}})$ and the total demand $(K_d^{\mathrm{die}})$, and compute it by

$$s_d^{\mathrm{pow}} = \frac{\int\int K_d^{\mathrm{die}}(x,y)\,dx\,dy}{\int\int \sum_{j=1}^{d} P_j^{\mathrm{cell}}(x,y)\,dx\,dy}. \tag{16}$$

The potential $\Phi_d^{\mathrm{pow}}$ for $B_d^{\mathrm{pow}}$ is computed by solving Poisson's equation

$$\Delta\Phi_d^{\mathrm{pow}}(x,y) = -B_d^{\mathrm{pow}}(x,y). \tag{17}$$

The target point $\mathring{x}_i^{\mathrm{P}}$ of TSV $i$ is computed by

$$\mathring{x}_i^{\mathrm{P}} = x_i' - \frac{\partial}{\partial x}\Phi_d^{\mathrm{pow}}(x,y)\Big|_{(x_i',y_i')} \tag{18}$$

where $x_i'$ is the x-position of TSV $i$ on die $d$ from the last iteration. We set spring constant $\mathring{w}_{\mathrm{x},i}^{\mathrm{P}}$ to $1/N_d^{\mathrm{TSV}}$, where $N_d^{\mathrm{TSV}}$ is the total number of TSVs on die $d$. Therefore, the power density-based spring for each TSVs has the same strength.

If a TSV $i$ sits on top of the hill of the potential $\Phi_d^{\mathrm{pow}}$, the target point $\mathring{x}_i^{\mathrm{P}}$ is the same as the x-position of the TSV from the last iteration $x_i'$ because of zero gradient in (18). That means we do not have to move this TSV because the thermal conductivity and power density of the region are already balanced.

*3) Balancing the Forces:* We balance the new forces against $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ because $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ is the main force that moves cells and TSVs. First, we scale the new forces so that they have the same magnitude as $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$. Then, we apply weighting constants to $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$, $\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}}$, and $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$ so that we can control their contribution to the total force.

First, to scale $\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}}$ to $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$, we normalize $P_d^{\mathrm{cell}}$, the demand for $B_d^{\mathrm{cond}}$ in (8), to $D_d^{\mathrm{cell}}$ by a scaling factor $s_d^{\mathrm{PD}}$ defined as

$$s_d^{\mathrm{PD}} = \frac{\int\int D_d^{\mathrm{cell}}(x,y)\,dx\,dy}{\int\int P_d^{\mathrm{cell}}(x,y)\,dx\,dy}. \tag{19}$$

Then, we replace $P_d^{\mathrm{cell}}$ in (8) and (9) by $s_d^{\mathrm{PD}} \cdot P_d^{\mathrm{cell}}$. Second, to scale $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$ to $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$, we normalize $K_d^{\mathrm{die}}$, the demand for $B_d^{\mathrm{pow}}$ in (15), to $D_d^{\mathrm{cell}}$ by a scaling factor $s_d^{\mathrm{KD}}$ defined as

$$s_d^{\mathrm{KD}} = \frac{\int\int D_d^{\mathrm{cell}}(x,y)\,dx\,dy}{\int\int K_d^{\mathrm{die}}(x,y)\,dx\,dy}. \tag{20}$$

Then, we replace $K_d^{\mathrm{die}}$ in (15) and (16) by $s_d^{\mathrm{KD}} \cdot K_d^{\mathrm{die}}$.

We scale both $\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}}$ and $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$ to $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ based on $D_d^{\mathrm{cell}}$, not on the gradient of $\Phi_d$ because of the stability issue. After normalizing $P_d^{\mathrm{cell}}$ and $K_d^{\mathrm{die}}$ to $D_d^{\mathrm{cell}}$ as shown in (19) and (20), the magnitude of $B_d^{\mathrm{cond}}$ and $B_d^{\mathrm{pow}}$ and gradient of their potential are properly normalized. At an equilibrium, a small magnitude of the gradients results in a small magnitude of $\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}}$ and $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$. If we scale $\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}}$ and $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$ to $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ based on the gradient of $\Phi_d$ instead, the magnitude of the gradient of potential of $B_d^{\mathrm{cond}}$ and $B_d^{\mathrm{pow}}$ would be exaggerated after the normalization, which in turn causes instability.

In summary, $\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}}$ moves cells in such a way that high-power density flows through the paths with high-thermal conductivity to heatsink. In addition, $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$ moves TSVs in such a way that each TSV establishes a heat path for the high-power cells in the same die and the dies below. Our overall force equation is as follows:

$$\mathbf{f}_{\mathrm{x}} = \mathbf{f}_{\mathrm{x}}^{\mathrm{net}} + \mathbf{f}_{\mathrm{x}}^{\mathrm{hold}} + (1-\alpha)\mathbf{f}_{\mathrm{x}}^{\mathrm{den}} + \alpha(\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}} + \mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}) = \mathbf{0}. \tag{21}$$

By increasing $\alpha$, the forces $\mathbf{f}_{\mathrm{x}}^{\mathrm{cond}}$ and $\mathbf{f}_{\mathrm{x}}^{\mathrm{pow}}$ dominate the movement of cells and TSVs for more thermal optimization. The impact of $\alpha$ is studied in Section V-B. Because cells and TSVs are moved simultaneously, they may chase each other around. In general, the effect, however, is minimal because $\mathbf{f}_{\mathrm{x}}^{\mathrm{den}}$ still dominates in the early placement iterations. The effect becomes noticeable in the late placement iterations only when the weighting factor $\alpha$ is high as revealed in Fig. 10.

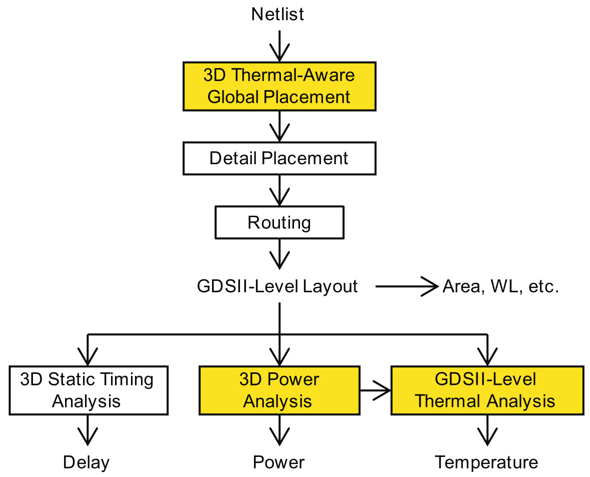

Fig. 7. Evaluation flow for temperature-aware 3-D IC global placement.

## IV. Evaluation Flow

In this section, we present our framework to evaluate the impact of TSVs on temperature of 3-D ICs. The main components of the framework are power analysis and GDSII-level thermal analysis for 3-D ICs. The presented evaluation flow allows us to evaluate the effectiveness of our 3-D temperature-aware global placement algorithms in reducing temperature. The result of our study is analyzed, and reported in detail in Section V.

Our evaluation flow for temperature-aware 3-D IC global placement is shown in Fig. 7. After obtaining 3-D temperature-aware global placement result, we perform detail placement and detail routing. We report traditional metrics, e.g., area and routed wirelength, of the final GDSII-level layout. We then perform 3-D static timing analysis, power analysis, and GDSII-level thermal analysis to report delay, power, and temperature, respectively. We explain how we use Cadence SoC Encounter and Synopsys PrimeTime PX for accurate power analysis for 3-D ICs in Section IV-A. We explain how we perform thermal

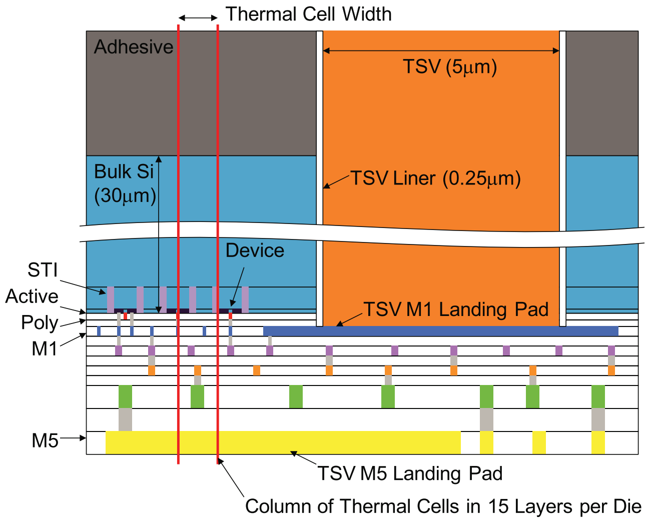

Fig. 8. Analyzed structure of a TSV-based 3-D IC. Each die is modeled with 15 layers of different materials. The entire 4-die structure contains 62 layers.

analysis from GDSII-level layouts of 3-D ICs by using Ansys FLUENT together with our layout analyzer in Section IV-B. Note that the result from power analysis needs to be presented to GDSII-level thermal analysis because logic cell power is the heat source in 3-D ICs during thermal analysis.

### A. Power Analysis for 3-D ICs

The power analysis flow for 3-D ICs developed in this paper starts by obtaining the layout of all dies in a 3-D IC in DEF or GDSII format. Next, we feed them to Cadence SoC Encounter to extract parasitic resistance and capacitance in SPEF format. We generate a separate SPEF file for parasitic resistance and capacitance of TSVs. The top-level verilog connects the verilog of all dies together, and the connection of all dies inside this top-level verilog represents TSVs. The switching activity of all logic cells in the whole design can be obtained by propagating switching probability, as well as static state probability, from all primary inputs into all nets of the design. Additional accuracy can be gained by performing functional simulation of the whole design. Finally, we use PrimeTime PX to perform static power analysis, and reports power dissipation of each logic cell. By stitching all the dies in this method, the parasitic resistance and capacitance of TSVs and wires running across dies also account for the total power of the 3-D IC.

### B. GDSII-Level Thermal Analysis

Steady-state temperature of a point $\mathbf{p} = (x, y, z)$ inside a 3-D structure can be obtained by solving the heat equation

$$\nabla \cdot (k(\mathbf{p})\nabla T(\mathbf{p})) + S_{\mathrm{h}}(\mathbf{p}) = 0 \qquad (22)$$

where $k$ is thermal conductivity in W/m · K, $T$ is temperature in K, and $S_{\mathrm{h}}$ is volumetric heat source in W/m$^3$. This model can be implemented by meshing analyzed structure of a 3-D IC into elements as shown in Fig. 8. Each element, called a thermal cell, is a volume of specific width and height, and its thickness is the same as each physical layer inside the 3-D IC.

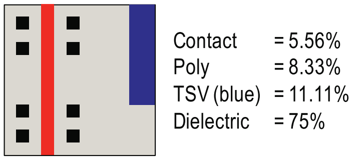

Fig. 9. Material composition inside a thermal cell.

To solve (22), boundary conditions must be given on the six surfaces of a 3-D chip stack. Generally, a 3-D chip stack is very thin and flat, and packaged inside molding materials, which are not good thermal conductor. The majority of heat flows from the stack toward the heatsink. Therefore, we apply adiabatic boundary condition on bottom and four sides of the stack, and apply convective boundary condition on the top side, which is the heatsink.

The thermal analysis flow developed in this paper starts by presenting the layout of all dies in a 3-D IC in GDSII format and power dissipation of each logic cell to the layout analyzer that we develop for this paper. The position of all TSVs is also presented to our layout analyzer so that all TSV related elements, e.g., landing pad and liner, are included into consideration. Our layout analyzer automatically generates meshed structure of the 3-D IC along with thermal conductivity and volumetric heat source of each thermal cell. The thermal conductivity is computed from all the structures (e.g., bulk Si, metal wires, and TSVs) extracted from the layout, and the volumetric heat source is computed from the power dissipation and position of each logic cell.

A thermal cell can be composed of several different materials, for example, polysilicon, tungsten in vias, copper in TSVs, and dielectric (see Fig. 9). With sufficiently fine thermal cell size, equivalent thermal conductivity based on thermal resistive model can be used [13]. In theory, if a thermal cell size is very small, material inside it is homogeneous, and its thermal conductivity is isotropic. However, using small cell size requires high-computing resource and long run time. For practical purpose, large thermal cell size can be used. Because of typical structural geometries found in GDSII layouts, thermal conductivity of each thermal cell is anisotropic. Vertical thermal conductivity $k_{\mathrm{ver}}$ and lateral thermal conductivity $k_{\mathrm{lat}}$ of a thermal cell consisting of $N$ materials can be computed from

$$k_{\mathrm{ver}} = r_1 \cdot k_1 + r_2 \cdot k_2 + \cdots + r_N \cdot k_N \qquad (23)$$

$$1/k_{\mathrm{lat}} = r_1/k_1 + r_2/k_2 + \cdots + r_N/k_N \qquad (24)$$

where $r_i$ is the ratio of material $i$ volume to thermal cell volume, and $k_i$ is the thermal conductivity of material $i$. Our layout analyzer computes $r_i$ directly from GDSII layout of all dies in the 3-D chip stack.

From the power dissipation and position of each logic cell, we can compute total power dissipated inside a thermal cell $P_{\mathrm{cell}}$. Then, volumetric heat source $S_{\mathrm{h}}$ can be computed from

$$S_{\mathrm{h}} = \frac{P_{\mathrm{cell}}}{W_{\mathrm{cell}} \cdot H_{\mathrm{cell}} \cdot T_{\mathrm{cell}}} \qquad (25)$$

where $W_{\mathrm{cell}}$, $H_{\mathrm{cell}}$, and $T_{\mathrm{cell}}$ are width, height, and thickness of the thermal cell, respectively.

TABLE II

BENCHMARK CIRCUITS

| Ckt. | #Gates | #TSVs | Util. | Footprt (mm$^2$) | Profile |
|---|---|---|---|---|---|
| ckt1 | 119,040 | 5,725 | 0.66 | 0.50 × 0.50 | Data encryption |
| ckt2 | 191,420 | 24,540 | 0.63 | 0.90 × 0.90 | Graphic accelerator |
| ckt3 | 280,933 | 17,362 | 0.49 | 0.98 × 0.98 | Video compression |
| ckt4 | 383,329 | 17,436 | 0.53 | 1.04 × 1.04 | Signal processing |
| ckt5 | 644,357 | 15,024 | 0.53 | 1.16 × 1.16 | Image encoder |

We solve (22) by using Ansys FLUENT, a commercial tool. Meshed structure generated from our layout analyzer can be presented to FLUENT directly. However, $k_{\text{ver}}$, $k_{\text{lat}}$, and $S_{\text{h}}$ need to be presented to FLUENT through user defined functions because they vary with thermal cell position. Finally, with the boundary conditions described earlier, we can run FLUENT to obtain steady-state temperature of all positions inside a 3-D chip stack.

## V. EXPERIMENTAL RESULTS

We use 45-nm technology from FreePDK45 for our experiments. TSV diameter is $5\,\mu$m, and the landing pad width is $7\,\mu$m. TSV liner thickness is 250 nm [14]. We use copper TSVs with SiO$_2$ liner [14] and 2.6-$\mu$m-thick benzocyclobutene bonding adhesive [1] for our experiments. Each die in the 3-D chip stack is thinned to $30\,\mu$m except that the topmost die, which is attached to heatsink, retains its thickness at $530\,\mu$m. The ambient temperature on top of the heatsink is 300 K. The TSV parasitic resistance and capacitance are $0.1\,\Omega$ and 125 fF, respectively. We base all our experiments on 4-die chip stacks.

We use IWLS 2005 benchmark designs. Our designs are also from www.opencores.org that provides RTL codes for "real" industry designs. We believe these are more practical than the gate/block-level netlists (= bare bone structure) used in most partitioning and placement papers. We use these designs so that we can perform detailed routing, obtain GDSII layouts, and perform sign-off thermal analysis, all based on a real 45 nm library. We synthesize the circuits using Synopsys Design Compiler to obtain gate-level netlist, and use the target clock period of each circuit when performing all analyses. The benchmark characteristics are listed in Table II. The numbers of TSVs are based on partitioning results from our own implementation of [6]. We use the same die partitioning results for all algorithms for fair comparison in Section V-C. Because [6] does not consider TSV area, it inserts high number of TSVs, resulting in low-placement utilization. We made our best effort to reduce the number of TSVs to improve utilization; however, even we set interlayer via coefficient (the parameter in [6] that controls the number of TSVs) to a high value, the number of TSVs was still high.

We do not optimize the circuits after placement because buffers and sized gates can change power profile, thus affecting temperature. The results reported in this paper are from commercial tools. We use Cadence Encounter to route the layouts, Synopsys PrimeTime to analyze timing and power, and Ansys FLUENT to analyze temperature. The thermal results presented in this section are performed at the same high-accuracy level. The thermal cell size is $5\,\mu$m, and we run Ansys FLUENT in double-precision mode until the solution converges with the residual less than $1 \times 10^{-16}$. We report all our temperature results in terms of the increase from the ambient temperature measured at the top of the heatsink.

TABLE III

ROUTED WIRELENGTH, LONGEST PATH DELAY, POWER OF PLACEMENTS WITH UNIFORM [10], AND NONUNIFORM [10] TSV POSITION

| | Uniform | | | Non-uniform | | |
|---|---|---|---|---|---|---|
| | rWL | $\mathrm{D_{max}}$ | P | rWL | $\mathrm{D_{max}}$ | P |
| Ckt. | (m) | (ns) | (W) | (m) | (ns) | (W) |
| ckt1 | 3.897 | 5.320 | 0.752 | 3.014 | 4.836 | 0.728 |
| ckt2 | 11.718 | 16.510 | 2.661 | 7.744 | 13.694 | 2.463 |
| ckt3 | 13.532 | 8.814 | 2.353 | 9.326 | 6.535 | 2.288 |
| ckt4 | 19.355 | 20.788 | 2.710 | 12.457 | 12.515 | 2.640 |
| ckt5 | 22.708 | 19.772 | 3.209 | 18.711 | 13.798 | 3.122 |
| ratio | 1.405 | 1.350 | 1.039 | 1.000 | 1.000 | 1.000 |

TABLE IV

TEMPERATURE (°C) OF PLACEMENTS WITH UNIFORM [10] AND NONUNIFORM [10] TSV POSITION. ($\Delta\mathrm{T_{ja}} = \mathrm{T_{ja,max}} - \mathrm{T_{ja,min}}$)

| | Uniform | | | Non-uniform | | |
|---|---|---|---|---|---|---|
| Ckt. | $\mathrm{T_{ja,max}}$ | $\Delta\mathrm{T_{ja}}$ | $\mathrm{T_{ja,ave}}$ | $\mathrm{T_{ja,max}}$ | $\Delta\mathrm{T_{ja}}$ | $\mathrm{T_{ja,ave}}$ |
| ckt1 | 71.55 | 17.60 | 64.50 | 74.13 | 18.33 | 63.98 |
| ckt2 | 101.14 | 47.14 | 69.41 | 94.41 | 50.19 | 64.78 |
| ckt3 | 70.38 | 31.01 | 55.06 | 80.09 | 42.81 | 55.48 |
| ckt4 | 64.91 | 18.76 | 54.32 | 75.98 | 38.01 | 55.16 |
| ckt5 | 66.77 | 35.40 | 53.13 | 75.24 | 39.32 | 54.50 |
| ratio | 1.000 | 1.000 | 1.000 | 1.081 | 1.325 | 0.995 |

### *A. Impact of TSV Density Uniformity*

In this experiment, we show how TSV density uniformity impacts thermal profile. Our two baseline 3-D placements are wirelength-driven placement with uniform TSV position [10] and wirelength-driven placement with nonuniform TSV position [10]. First, we obtain both baseline placements using our own implementation of [10].[5] Then, we perform power and thermal analyses on both placement results. The routed wirelength, longest path delay, and power are shown in Table III, and temperatures are shown in Table IV. Although the placement with nonuniform TSV position has shorter wirelength, better timing, and lower power than the placement with uniform TSV position, its temperature, especially the thermal variation, is worse. Both the nonuniform power density and the nonuniform thermal conductivity, caused by the nonuniform distribution of TSVs in the 3-D chip stack, contribute to the problem. In the placement with nonuniform TSV position, we observe that the area with high-TSV density has low-power density and low temperature, vice versa. These two opposite trends are responsible for high-thermal variation.

### *B. Temperature-Wirelength Trade-off*

Our thermal-coupling-aware placement algorithm provides an efficient way to explore temperature-wirelength trade-

[5] It is possible to use well-known academic 2-D placers such as CAPO, NTUplace, mPL, FastPlace, etc. [15], to obtain uniform TSV placement. However, these placers need to be extended to handle preplaced TSVs. In addition, these preplaced TSVs need to be assigned to 3-D nets before the gate placement starts [10]. Lastly, thermal objective needs to be incorporated in these placers.

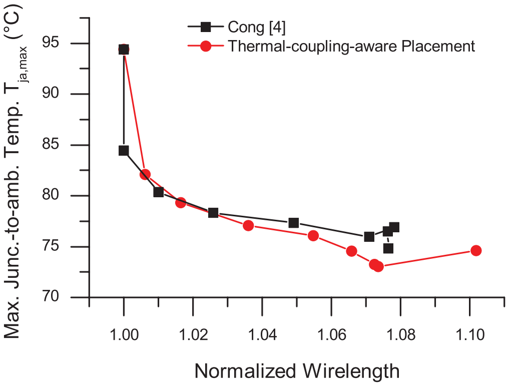

Fig. 10. Temperature-wirelength trade-off.

off. We study, the temperature-wirelength trade-off in this experiment. By increasing the weighting constant $\alpha$ in (21), the placer increases the magnitude of $\mathbf{f}_x^{\text{cond}}$ and $\mathbf{f}_x^{\text{pow}}$ while decreasing $\mathbf{f}_x^{\text{den}}$, i.e., trading wirelength for temperature. The temperature-wirelength tradeoff for ckt2 is shown in Fig. 10. We also implemented the placer from [7] ourselves and show its trade-off curve in Fig. 10. A weighting constant $\beta$ is used in [7]. The range of both $\alpha$ and $\beta$ in this experiment is [0, 0.5].

With $\alpha = \beta = 0$, both placers perform as a wirelength-driven placer (= left-most points). As we increase $\alpha$ and $\beta$, temperature decreases while wirelength increases. We observe that as $\alpha$ and $\beta$ increase, our thermal coupling-aware placer outperforms [7]: our placer has shorter routed wirelength at the same temperature, and has lower temperature at the same wirelength. We also observe that [7] shows convergence problem with large $\beta$ values. When [7] moves a high-power cell into a bin, it moves cells out of other bins in the dies above or below, resulting in potential wirelength increase and convergence problem as discussed in [7]. In addition, [7] does not consider vertical alignment of TSVs so that even if it moves high-power cells into a bin with many TSVs, the heat captured in the bin may not be easily dissipated vertically to the heatsink. Our algorithms overcome these limitations.

### *C. Comparison With State-of-the-Art*

We compare our temperature-aware global placement algorithms with the following recent state-of-the-art temperature-aware placers. This task is challenging due to the discrepancy among the settings and assumptions made in each work. However, we made our best effort to provide fair and meaningful comparison, including in-depth discussions with the authors. We reimplemented all the works after contacting all the original authors. We reimplemented [5] and [9] starting from our own placer [10] because they all use quadratic wirelength model. We reimplemented [6] starting from our own placer [16] because both of them are partitioning-based placer. We applied the temperature-aware concept of [7] to our own placer [10], but did not reimplement the LSE wirelength model and multilevel framework.

Reference [5] (force-directed placer): In this paper, thermal analysis is performed at the beginning of every global placement iteration. The thermal gradient obtained from the analysis is used to compute repulsive force, which moves logic cells from high-temperature area toward low-temperature area. We implement our own version of this paper by calling Ansys FLUENT from inside our placer, and combining scaled thermal gradient into density-based force $\mathbf{f}_x^{\text{den}}$.

Reference [9] (force-directed placer): Instead of moving logic cells based on placement area density, it moves logic cells based on placement power density. Therefore, logic cells are spread according to their power dissipation, and logic cells with high-power dissipation occupy more space than logic cells with low-power dissipation, leading to uniform power density and thermal profile across the die. We implement our own version of this paper.

Reference [6] (partitioning-based placer): In this paper, logic cells are partitioned into placement area and different dies based on the switching activity and parasitic capacitance of connecting wires and TSVs. We perform global routing to determine the position of TSVs as proposed in [16] after performing global placement using our own implementation of [6].

Reference [7] (analytical placer): We implement this method by balancing the power density combined across dies in vertical direction against the TSV density and solving the density for potential function. The gradient of potential is used to compute a force to move cells and TSVs to maintain the balance. The force is added to $\mathbf{f}_x^{\text{den}}$ with a user-defined parameter $\beta$ to provide temperature-wirelength trade-off similar to the work.

Table V shows the routed wirelength, delay, power, and temperature comparison based on the GDSII layouts we build using these placers. The wirelength, delay, and power values are normalized to the wirelength-driven nonuniform TSV placement [10] shown in Table III. The temperature values are normalized to the wirelength-driven uniform TSV placement [10] shown in Table IV. Recall that nonuniform placer achieves high-quality wirelength, delay, and power results while uniform placer leads to high-quality temperature values. We increase $\alpha$ for our thermal coupling-aware placement and $\beta$ for our implementation of [7] from 0 to 0.5. When the temperature does not decrease much with significant increase in wirelength, we report the result for each method at that $\alpha$ and $\beta$. The range of $\alpha$ and $\beta$ for the results reported in Table V is [0.1, 0.4].

First, we observe that [5] produces comparable wirelength, delay, and power results to nonuniform TSV placer [10]. In case of temperature, [5] obtains worse result compared with uniform TSV placer [10]. We tried increasing the magnitude of thermal-gradient-based force, and found large increase in wirelength without much additional temperature improvement. Moving cells out of a high-temperature area on a die may not reduce temperature if the high temperature is a result from thermal coupling with other dies. Also, without considering TSV thermal properties during thermal analysis, the thermal gradient does not capture the impact of TSVs on temperature accurately, thereby misguiding the placement. Second, we see that [9] obtains wirelength and delay results that are significantly worse than nonuniform TSV placer. This is mainly

TABLE V

COMPARISON WITH STATE-OF-THE-ART TEMPERATURE-AWARE PLACERS [5]–[7], [9], [10]. OUR PLACERS ARE TSA (TSV SPREAD AND ALIGNMENT) AND CA (COUPLING-AWARE PLACEMENT). THE ROUTED WIRELENGTH, DELAY, AND POWER VALUES ARE NORMALIZED TO THE NONUNIFORM TSV PLACEMENT [10] SHOWN IN TABLE III. THE TEMPERATURE VALUES ARE NORMALIZED TO THE UNIFORM TSV PLACEMENT [10] SHOWN IN TABLE IV

| | routed wirelength (m) | | | | | | longest path delay (ns) | | | | | | power consumption (W) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Ckt. | [5] | [9] | [6] | [7] | TSA | CA | [5] | [9] | [6] | [7] | TSA | CA | [5] | [9] | [6] | [7] | TSA | CA |
| ckt1 | 3.046 | 3.109 | 3.784 | 3.240 | 3.250 | 3.133 | 4.935 | 4.796 | 5.128 | 5.067 | 4.786 | 4.871 | 0.729 | 0.734 | 0.776 | 0.736 | 0.736 | 0.732 |
| ckt2 | 7.740 | 8.780 | 14.924 | 8.349 | 7.892 | 8.314 | 13.679 | 15.004 | 15.231 | 14.416 | 13.588 | 14.785 | 2.463 | 2.548 | 2.564 | 2.521 | 2.487 | 2.523 |
| ckt3 | 9.347 | 10.544 | 16.028 | 10.706 | 10.355 | 10.261 | 6.567 | 6.797 | 7.865 | 7.276 | 6.530 | 6.906 | 2.290 | 2.331 | 2.351 | 2.318 | 2.306 | 2.321 |
| ckt4 | 12.480 | 13.902 | 19.871 | 15.234 | 14.901 | 14.545 | 12.518 | 12.695 | 16.158 | 13.609 | 13.695 | 13.113 | 2.640 | 2.671 | 2.737 | 2.682 | 2.672 | 2.675 |
| ckt5 | 18.869 | 21.482 | 27.649 | 20.125 | 19.845 | 19.994 | 13.931 | 16.427 | 15.649 | 13.674 | 13.799 | 14.664 | 3.127 | 3.194 | 3.255 | 3.166 | 3.130 | 3.156 |
| ratio | **1.005** | 1.112 | 1.595 | 1.120 | 1.093 | 1.090 | **1.007** | 1.066 | 1.160 | 1.058 | 1.015 | 1.051 | **1.001** | 1.019 | 1.043 | 1.015 | 1.009 | 1.014 |

| | max junc.-to-amb. temp, $\mathrm{T_{ja,max}}$ (°C) | | | | | | temp difference, $\mathrm{T_{ja,max} - T_{ja,min}}$ (°C) | | | | | | average temp, $\mathrm{T_{ja,ave}}$ (°C) | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Ckt. | [5] | [9] | [6] | [7] | TSA | CA | [5] | [9] | [6] | [7] | TSA | CA | [5] | [9] | [6] | [7] | TSA | CA |
| ckt1 | 72.48 | 73.12 | 82.86 | 70.69 | 70.85 | 70.41 | 16.29 | 14.94 | 28.12 | 14.69 | 15.55 | 14.16 | 63.80 | 63.70 | 69.52 | 63.32 | 63.27 | 63.35 |
| ckt2 | 91.70 | 74.21 | 101.00 | 76.89 | 100.19 | 73.05 | 46.96 | 15.15 | 51.16 | 22.39 | 53.87 | 17.15 | 64.81 | 66.84 | 69.36 | 66.07 | 65.14 | 66.14 |
| ckt3 | 77.74 | 64.39 | 69.80 | 66.34 | 72.41 | 65.60 | 39.89 | 19.68 | 28.69 | 23.82 | 33.65 | 22.97 | 55.41 | 55.49 | 55.97 | 54.53 | 54.14 | 55.08 |
| ckt4 | 73.79 | 62.43 | 80.11 | 60.14 | 65.50 | 59.31 | 35.46 | 16.69 | 39.76 | 15.87 | 21.83 | 14.27 | 55.07 | 54.35 | 60.42 | 53.91 | 53.63 | 53.85 |
| ckt5 | 74.86 | 79.22 | 76.25 | 61.95 | 64.45 | 61.60 | 38.08 | 36.39 | 38.02 | 23.77 | 33.07 | 24.53 | 54.51 | 55.08 | 57.97 | 53.22 | 51.91 | 52.90 |
| ratio | 1.056 | 0.964 | 1.105 | 0.909 | 0.997 | **0.895** | 1.235 | 0.744 | 1.360 | 0.719 | 1.042 | **0.673** | 0.994 | 0.999 | 1.059 | 0.984 | **0.973** | 0.984 |

because it moves logic cells based only on power density. However, this move helps reduce maximum temperature and thermal variation inside the 3-D chip stack significantly. Although it attempts to spread power over placement area, we observe that TSVs obstruct this effort frequently.

Third, the routed wirelength and delay of results from [6] are worse than all other placers. The main reason is that [6] does not consider TSV area during placement. Thus, the TSVs inserted during routing affects the placement quality significantly. The maximum temperature, thermal variation, and average temperature are also worse than uniform TSV placer. The router tends to insert TSVs in the middle of the die to minimize wirelength, leaving low-thermal conductivity at chip corners, thus high temperature. Fourth, although the wirelength of result from [7] is worse than other placers, temperature improvement is among the best. Because the algorithm consider the impact of TSV on chip area and temperature, it utilizes TSVs more effectively to help improve temperature results.

Fifth, we observe that our TSA achieves comparable delay and power results at the cost of wirelength degradation compared with nonuniform placer. In case of temperature, TSA obtains better average temperature than uniform TSV and comparable maximum temperature and temperature difference. But, the wirelength of TSA method is significantly better than that of uniform TSV placer. These results show that our TSA method is better in reducing wirelength while optimizing temperature compared with uniform TSV placer.

Lastly, our thermal CA achieves the best temperature results among all placers [5]–[7], [9], including uniform TSV placer [10]. In particular, our CA method outperforms uniform TSV placer by 10% and 33% in terms of maximum temperature and temperature difference. CA obtains 9% worse wirelength and 5% worse delay results compared with nonuniform TSV placer, but CA is among the best in terms of wirelength and delay among other placers [5]–[7], [9]. The power overhead is negligible. The TSVs in the placement by our CA method are not spread as evenly as our TSA placer and uniform TSV placer, but they are spread only sufficiently to help remove heat from the dies in the stack while maintaining high-quality wirelength. In addition, we observe that high-power logic cells are also placed effectively to dissipate heat using the nearby TSVs that are vertically aligned all the way to the heatsink.

TABLE VI

RUNTIME COMPARISON OF UNIFORM TSV PLACEMENT [10], NONUNIFORM TSV PLACEMENT [10], STATE-OF-THE-ART TEMPERATURE-AWARE PLACERS [5]–[7], [9] AND OUR PLACERS. OUR PLACERS ARE TSA (TSV SPREAD AND ALIGNMENT) AND CA (COUPLING-AWARE PLACEMENT)

| | runtime (min.) | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| Ckt. | [10] | [10] | [5] | [9] | [6] | [7] | TSA | CA |
| ckt1 | 13.04 | 11.07 | 19.89 | 25.78 | 31.15 | 21.00 | 9.06 | 24.04 |
| ckt2 | 62.96 | 52.38 | 75.50 | 96.81 | 49.49 | 67.73 | 52.68 | 99.56 |
| ckt3 | 45.05 | 42.46 | 78.09 | 127.29 | 65.10 | 95.70 | 53.35 | 102.40 |
| ckt4 | 74.88 | 58.25 | 102.87 | 231.77 | 88.01 | 262.18 | 90.59 | 244.48 |
| ckt5 | 169.10 | 229.08 | 293.14 | 388.36 | 165.47 | 652.01 | 168.51 | 423.03 |
| total | 365.04 | 393.24 | 569.50 | 870.01 | 399.22 | 1,098.62 | 374.18 | 893.52 |

### *D. Runtime Results*

The runtime of wirelength-driven placer with uniform TSV position [10], wirelength-driven placer with nonuniform TSV position [10], state-of-the-art temperature-aware placers [5]–[7], [9], and our placers are shown in Table VI. The runtime for [5] includes running power analysis and thermal analysis between iterations. The runtime for [7], [9] and our thermal coupling-aware placer includes running power analysis between iterations. The runtime of all temperature-aware placement algorithms is roughly in the same magnitude. Except for our TSA method, all other placement algorithms require power simulation (and thermal simulation in the case of [5]), resulting in larger runtime than [10].

Although Ansys FLUENT is integrated within our own implementation of [5], the runtime of our implementation of [5] is still similar to other approaches because of the following three reasons. First, we do not run thermal analysis (and power analysis) during the early iterations of global placement when cell overlap is very high. Second, as done in the original work [5], we do not include TSVs and metal wires in thermal analysis because their position is not conclusive during global placement. Third, we use a moderate thermal cell size of 20 $\mu$m for our implementation of [5]. Thus, these settings altogether reduce the runtime overhead of thermal analysis.

## VI. Conclusion

In this paper, we showed that temperature-aware placers must consider TSV thermal properties and die-to-die thermal coupling during placement. We presented two temperature-aware placement algorithms for 3-D ICs. TSVs are spread and aligned in the first algorithm. In the second algorithm, logic cells are moved based on the thermal conductivity to the heatsink, and TSVs are moved based on the power density of the neighboring dies. Experimental results show that our placers achieve the best temperature results among all placers used in our comparison. Our future directions include thermal-aware mixed-size placement for 3-D ICs with TSVs, and the consideration of temperature-dependent leakage power.

## References

[1] P. Leduc, F. De Crecy, M. Fayolle, B. Charlet, T. Enot, M. Zussy, *et al.*, "Challenges for 3D IC integration: Bonding quality and thermal management," in *Proc. IEEE Int. Interconnect Technol. Conf.*, Jun. 2007, pp. 210–212.

[2] C. C.-P. Chen, J.-L. Tsai, G. Chen, B. Goplen, H. Qian, Y. Zhan, *et al.*, "Temperature-aware placement for socs," *Proc. IEEE*, vol. 94, no. 8, pp. 1502–1518, Sep. 2006.

[3] J. Cong, G. Luo, J. Wei, and Y. Zhang, "Thermal-aware 3D placement via transformation," in *Proc. Asia South Pacific Design Autom. Conf.*, Jan. 2007, pp. 780–785.

[4] H. Yan, Q. Zhou, and X. Hong, "Thermal-aware in 3D ICs using quadratic uniformly modeling approach," *Integr. VLSI J.*, vol. 4, no. 2, pp. 175–180, Feb. 2009.

[5] B. Goplen and S. Sapatnekar, "Efficient thermal placement of standard cells in 3D ICs using a force directed approach," in *Proc. IEEE Int. Conf. Comput. Aided Design*, Nov. 2003, pp. 86–89.

[6] B. Goplen and S. Sapatnekar, "Placement of 3D ICs with thermal and interlayer via considerations," in *Proc. ACM Design Autom. Conf.*, Jun. 2007, pp. 626–631.

[7] J. Cong, G. Luo, and Y. Shi, "Thermal-aware cell and through-silicon-via co-placement for 3D ICs," in *Proc. ACM Design Autom. Conf.*, Jun. 2011, pp. 670–675.

[8] S. S. Sapatnekar, "Physical design automation challenges for 3D ICs," in *Proc. Int. Conf. IC Design Technol.*, 2006, pp. 1–5.

[9] B. Obermeier and F. M. Johannes, "Temperature-aware global placement," in *Proc. Asia South Pacific Design Autom. Conf.*, Jan. 2004, pp. 143–148.

[10] D. H. Kim, K. Athikulwongse, and S. K. Lim, "A study of through-silicon-via impact on the 3D stacked IC layout," in *Proc. IEEE Int. Conf. Comput. Aided Design*, Nov. 2009, pp. 674–680.

[11] P. Spindler, U. Schlichtmann, and F. M. Johannes, "Kraftwerk2—A fas force-directed quadratic placement approach using an accurate net model," *IEEE Trans. Comput. Aided Design Integr. Circuits Syst.*, vol. 27, no. 8, pp. 1398–1411, Aug. 2008.

[12] D. H. Kim, K. Athikulwongse, and S. K. Lim, "Study of through-silicon-via impact on the 3-D stacked IC layout," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 5, pp. 862–874, May 2013.

[13] C. Xu *et al.*, "Fast 3-D thermal analysis of complex interconnect structures using electrical modeling and simulation methodologies," in *Proc. IEEE Int. Conf. Comput. Aided Design*, Nov. 2009, pp. 658–665.

[14] G. Van der Plas, P. Limaye, A. Mercha, H. Oprins, C. Torregiani, S. Thijs, *et al.*, "Design issues and considerations for low-cost 3D TSV IC technology," in *IEEE Int. Solid-State Circuits Conf. Dig. Tech. Paper*, Feb. 2010, pp. 148–149.

[15] N. Viswanathan, C. Alpert, C. Sze, Z. Li, and Y. Wei, "The DAC 2012 routability-driven placement contest and benchmark suite," in *Proc. ACM Design Autom. Conf.*, Jun. 2012, pp. 774–782.

[16] M. Pathak, Y.-J. Lee, T. Moon, and S. K. Lim, "Through-silicon-via management during 3D physical design: When to add and how many?" in *Proc. IEEE Int. Conf. Comput. Aided Design*, Nov. 2010, pp. 387–394.

**Krit Athikulwongse** (S'04–M'13) received the B.Eng. and M.Eng. degrees from the Department of Electrical Engineering, Chulalongkorn University, Bangkok, Thailand, in 1995 and 1997, respectively, and the M.S. and Ph.D. degrees from the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA, in 2005 and 2012, respectively.

He was an Engineer with the Electricity Generating Authority of Thailand from 1998 to 2001. Since 2012, he has been a Researcher with the National Electronics and Computer Technology Center, Pathum Thani, Thailand. His current research interests include embedded systems, physical design, computer architecture, 3-D ICs, and VLSI design.

**Mongkol Ekpanyapong** (S'03–M'06–SM'11) received the B.Eng. degree from Chulalongkorn Univerisity, Bangkok, Thailand, in 1997, the M.Eng. degree from the Asian Institute of Technology, Pathum Thani, Thailand, in 2000, and the M.Sc. and Ph.D. degrees from the Georgia Institute of Technology, Atlanta, GA, USA, in 2003, and 2006, respectively.

He was a System Engineer with United Communication Network, Bangkok, from 1997 to 1998. From 2006 to 2009, he was a Senior Computer Architect with Intel Corporation, USA, Core 2 Architecture Design Team. He joined the School of Engineering and Technology, Asian Institute of Technology, in 2009, where he currently is an Assistant Professor. His current research interests include VLSI design, physical design automation, microarchitecture, compiler, and embedded systems.

**Sung Kyu Lim** (S'94–M'00–SM'05) received the B.S., M.S., and Ph.D. degrees from the Computer Science Department, University of California, Los Angeles (UCLA), CA, USA, in 1994, 1997, and 2000, respectively.

He joined the School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA, USA, in 2001, where he is currently an Associate Professor. He is the author of *Practical Problems in VLSI Physical Design Automation* (Springer, 2008). His current research interests include architecture, circuit, and physical design for 3-D ICs and 3-D system-in-packages.

Dr. Lim received the Design Automation Conference (DAC) Graduate Scholarship in 2003, the National Science Foundation Faculty Early Career Development (CAREER) Award in 2006, and the ACM SIGDA Distinguished Service Award in 2008. He was on the Advisory Board of the ACM Special Interest Group on Design Automation (SIGDA) from 2003 to 2008. He was an Associate Editor of the IEEE Transactions on Very Large Scale Integration (VLSI) Systems from 2007 to 2009. His papers were nominated for the Best Paper Award at ISPD in 2006, ICCAD in 2009, CICC in 2010, DAC in 2011, DAC in 2012, and ISLPED in 2012.