IPSJ Transactions on System LSI Design Methodology Vol. 3 2–18 (Feb. 2010)

Invited Paper

Advances and Challenges in 3D Physical Design

Jason Cong[†1,†2] and Guojie Luo[†1]

The task of 3D physical design is to map a circuit from a netlist (structural) representation into a geometric (physical) representation according to a specific 3D IC technology with multiple active device layers. This paper discusses the recent progress made on the major steps in 3D physical design, including 3D floorplanning, 3D placement, 3D routing and thermal through-silicon via (TS via) planning, and outlines the challenges ahead.

1. Introduction

The physical design process for 3D ICs is similar to that used for the traditional 2D physical design, in a sense that it transforms the circuit representation from a netlist into a geometric representation by the steps of floorplanning, placement and routing. While the multiple-layer metals have already had 3D structure in traditional ICs for interconnects, the 3D IC technologies allow multiple layers of logic devices to be integrated in the third dimension by bonding stacks of multiple "tiers" to form 3D chips. Each *tier*, which is similar to a traditional 2D IC, consists of one silicon layer and several metal layers, and different tiers are connected by through-silicon vias (TS via).

Figure 1 shows two examples of 3-tier 3D ICs in a cross-section view. The bottom tier, the middle tier and the top tier are labeled 1, 2, and 3, respectively. The physical layers are parallel to the (x, y) plane, and are bonded along the z-direction, where the darker shaded bands are dielectric layers, the lighter shaded bands are silicon layers, and the white bands are metal layers. The large rectangles vertically drilling through silicon layers represent TS vias, which connect logic gates on different silicon layers. The I/O ports open above the topmost layer. Figure 1 (a) presents a 3D IC by bonding three tiers in a back-to-face order, where the back side (the silicon layer) of the upper-level tier is bonded to the front side (the topmost metal layer) of the lower-level tier. Figure 1 (b) presents another 3D IC, where the middle tier is bonded face-to-face to the bottom tier, and the top tier is bonded face-to-back to the middle tier.

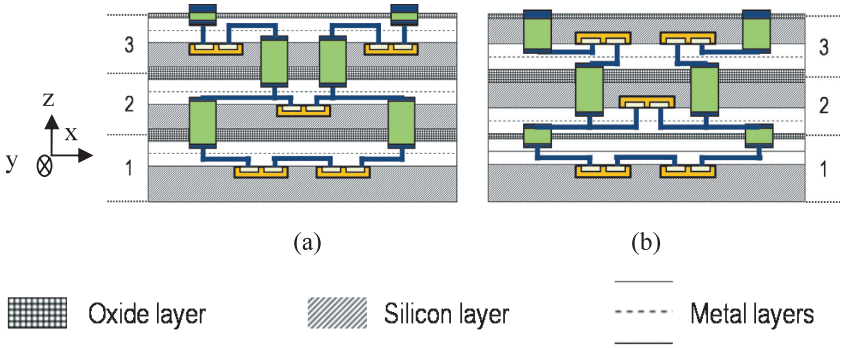


Fig. 1 Two examples of 3D ICs in a cross-section view.

The requirements on physical design tools to support 3D IC technologies come from several aspects [3),4),9)]. The latency and power are still important criterion, where the floorplanning and placer have to consider the timing and power characteristics of TS vias. The thermal issues in 3D ICs become critical: (1) The vertically stacked multiple layers of active devices cause a rapid increase in power density; (2) The thermal conductivity of the dielectric layers between the device layers is very low compared to silicon and metal. For instance, the thermal conductivity at the room temperature (300 K) for SiO_2 is 1.4 W/mK [27)], which is much smaller than the thermal conductivity of silicon (150 W/mK) and copper (401 W/mK). Therefore, the thermal issue needs to be considered during every step of the 3D physical design flow.

A reference 3D physical design flow is shown in **Fig. 2**, as developed in Refs. 14), 17). The 3D design database holds the necessary information for physical design tools, including the technology library (e.g., design rules, attributes of physical layers), cell/macro library, and the netlist. The netlist is transformed to a 3D geometric representation in the steps of 3D floorplanning, 3D placement and 3D

†1 Computer Science Department, University of California, Los Angeles
†2 California NanoSystems Institute

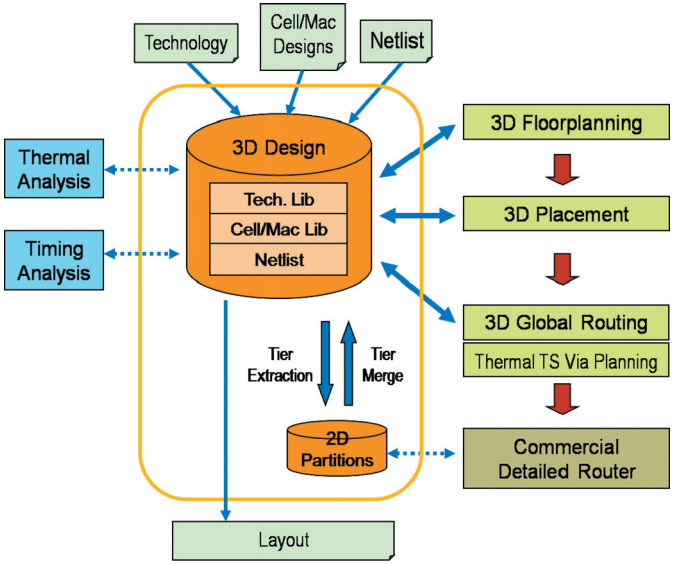


Fig. 2 Physical design flow for 3D ICs.

routing, and the thermal issues of 3D ICs are relieved by adopting the thermal TS vias [10),16),34),35)]. These steps form the main flow for the 3D physical design, which are covered in this paper. Please note that other supporting steps, such as power grid optimization [60)] and clock tree synthesis [39),44)], are also important for 3D physical design, but are not addressed due to page limitations.

In the remaining of this paper, we shall present the steps of 3D physical design in the reversed order starting with 3D routing and thermal TS via planning (Section 2). Then we present the problem formulations and algorithms of 3D placement (Section 3) and 3D floorplanning (Section 4). Finally, Section 5 concludes this paper and discusses future challenges.

2. 3D Routing and Thermal TSV Planning

Given the placement of every cell and every macro, either manually or automatically, 3D routing is used to connect all the cells and macros by metal wires, vias and TS vias according to the netlist information, without violating the design rules, under constraints like timing, crosstalk, temperature and yield.

2.1 Problem Formulation

The inputs of a 3D routing problem include:

- *Design rules*: They specify the minimum sizes and spacing of the metal wires, vias, and TS vias.
- *Netlist*: It specifies the connectivity of pins.
- *Pin locations*: The pins include the I/O pins of the top-level design, and the pins of all the cells and macros, the locations of which are determined after the 3D placement step.
- *Obstacles*: The pre-routed nets create obstacles, and the placed cells and macros create obstacles for TS vias.
- *Constraint-related parameters*: They include the electrical parameters, thermal parameters, yield parameters, etc., for various design constraints.

Two examples of the 3D routing resources are shown in **Fig. 3** (a) and Fig. 3 (b), which correspond to the 3D ICs in Fig. 1 (a) and Fig. 1 (b), respectively. The postfix of the layer names in Fig. 3 represents the tier in which this layer is located, and the prefix represents the layer type. The silicon layer is labeled with a prefix TSV, where the interconnect going through this layer is implemented by TS via. The metal layers are labeled with prefixes m1 and m2 with gray shading, and the via layer between them is labeled with v12. Although there are only two metal layers of each tier shown in these examples for the convenience of demonstration, more metal layers are manufacturable in 3D IC technologies. The interconnects are routed in orthogonal directions inside the metal layers, and they are routed in a vertical direction on the via layer and TS via layers.

The pins of cells and macros usually locate at the low-level metal layers, as in 2D ICs, and the I/O pins locate at the topmost layer in the 3D IC layer stack. Obstacles may exist at every layer, where the pre-routed nets create obstacles on the metal layers and via layers, and the cells and macros create obstacles on the TSV layers.

The 3D routing models are very similar to the 2D routing models with metal layers and via layers only, where the TS via layers can be viewed as special via layers. The major differences are that: (i) there are many more obstacles on the TS via layers than on the via layers, due to the placed cells and macros; (ii) the

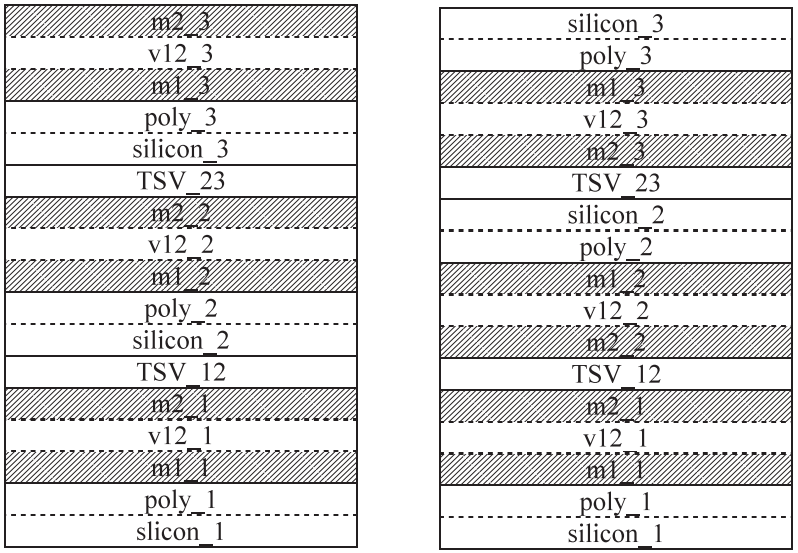


Fig. 3 Two examples of the 3D routing models.

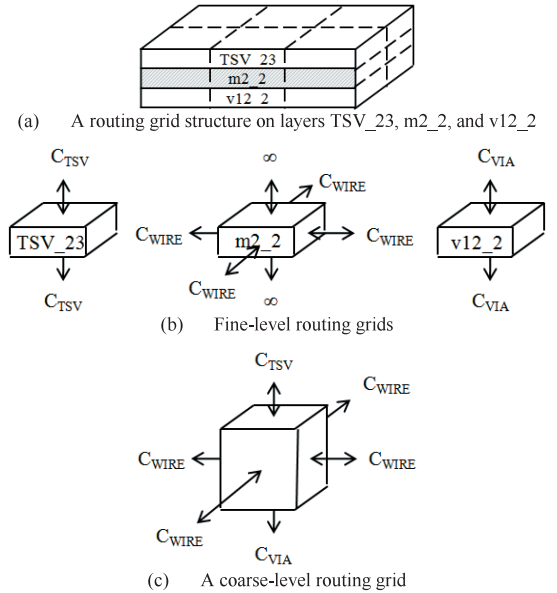


Fig. 4 Routing grid structure and switching box models.

minimum size and spacing rules of the TS via layers are much larger than those of the via layers; and (iii) there are tight thermal constraints. Clearly, the 3D routing problem is a generalized version of the routing problem for multi-metal layer 2D ICs.

Since the thermal issues are critical for 3D designs, the concept of the thermal TS via is proposed as an effective way to reduce temperature [10),34)]. The thermal TS via planning problems can be formulated as below: given a netlist, the place & route (P&R) region, and the thermal analysis model, find the location of thermal TS vias to satisfy the temperature constraint without violating the feasibility and degrading the quality of the 3D P&R results. The thermal TS via planning can be performed before routing, in routing, or after routing.

2.2 3D Global Routing Algorithms

Researchers began to investigate the 3D channel routing problems [24),48)] in the early nineties purely out of theoretical interest. In recent years, the area routers dominate when multiple metal layers are available for routing, and so do the 3D routers, where the routing problems can be modeled as in Fig. 3. The 3D routing process can be divided into global routing and detailed routing stages. In this paper we focus on the global routing problem, because once the TS via positions are determined, we can take advantage of the existing 2D detailed routers to complete the detailed routings tier by tier.

During 3D global routing, a grid structure is imposed on the routing layers, as shown in **Fig. 4** (a). Each grid is modeled by a switching box with six capacities along the x-axis, y-axis, and z-axis in both directions, where the capacities are measured by the number of interconnects allowed. A single layer of grids can model a physical layer at a fine level [15)], as shown in Fig. 4 (b). And it can also model a sequence of physical layers, which usually form a tier, at a coarse level [58)], as shown in Fig. 4 (c). For example, if a layer of grids model a physical layer as in Fig. 4 (b), the capacities can be computed in this way: (1) the capacities of a grid on the metal layers (C_{WIRE}) along the x-axis and y-axis are computed

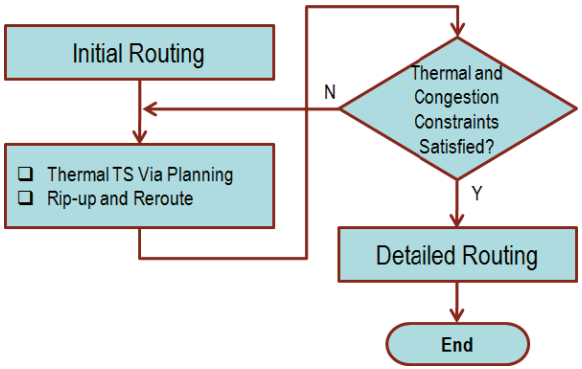


Fig. 5 The flat global routing scheme.

according to the obstacles inside that grid, and the capacities along the z-axis are infinite (eventually limited by the neighboring layers); (2) the capacities of a grid on the via layers and TS via layers along the x-axis and y-axis are zero, and the capacities along the z-axis (C_{TSV} or C_{VIA}) are computed according to the obstacles inside that grid.

Roughly speaking, the 3D global routing problems are very similar to the 2D routing problems, because the multiple metal layers already create some 3D structures. The additional considerations in 3D routing problems include: (1) the solution space is larger than that of 2D routing problems because more layers are available; (2) the pins locate on more layers in 3D routing problems than in 2D routing problems, where the pins only locate on a few metal layers close to the silicon layer; (3) the TS vias create blockages and consume routing resources to go through silicon layers; (4) thermal optimization is available based on the idea of thermal TS vias.

The global routing flow can be implemented either in a flat global routing scheme (**Fig. 5**), or in a multilevel global routing scheme (**Fig. 6**). In the flat global routing scheme [58], the initial 3D global routing consists of the signal TS via planning and the wire routing. After the initial routing, iterations of thermal TS via planning and rip-up and reroute are performed to meet the congestion constraints and the thermal constraints.

In the multilevel global routing scheme [15], a V-cycle consists of a downward

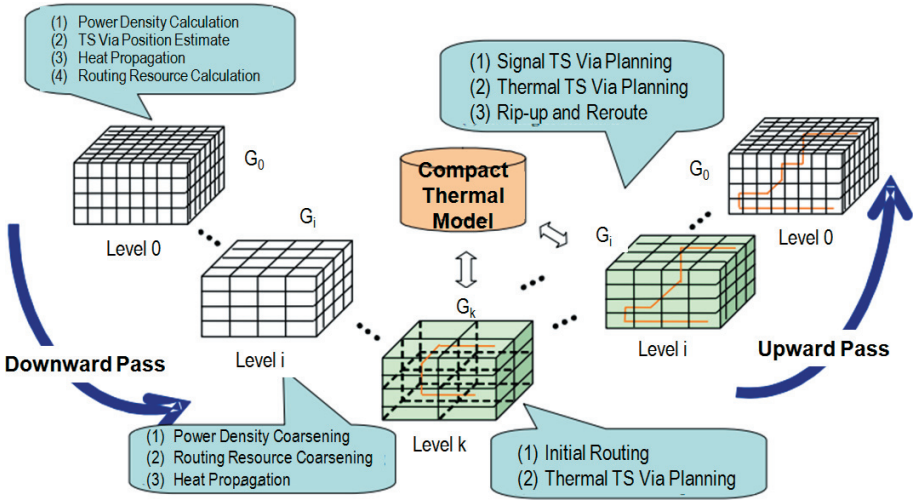


Fig. 6 The multilevel global routing scheme [15].

pass and an upward pass. In the downward pass, the coarse-level 3D global routing problems are constructed from the fine-level problems by estimating the coarse-level routing resources and the thermal-related information. These problems are solved in the upward pass from the coarsest level to the finest level, where the local nets are routed and the routing of global nets is refined. At the coarsest level, an initial 3D global routing and the thermal TS via planning are performed. Then the coarse-level routing results are projected to a finer-level problem, and the finer-level problem is solved by completing the signal TS via planning, the thermal TS via planning and the rip-up and reroute. After the 3D global routing is done and the TS via locations are determined at the finest level, 2D detailed routing is performed tier-by-tier to finish the 3D routing process.

In both the flat scheme and the multilevel scheme, the rip-up and reroute is usually for wire routings, thus, the techniques for 2D rip-up and reroute can be adopted. In the following subsections, we shall focus on the key components for 3D global routing algorithms, including initial 3D global routing (Section 2.2.1), signal TS via planning (Section 2.2.2) and thermal TS via planning (Section 2.2.3).

2.2.1 Initial 3D Global Routing

During the initial 3D global routing stage, the signal TS via locations can be determined before or during wire routing. The work in Ref. 58) plans TS vias before wire routing using three steps: routing congestion estimate, signal TS via planning, and 2D wire routing. The first step estimates the routing congestion by extending the L-Z shaped statistical routing model [51] for 3D global routing. Based on the congestion information, the second step plans the signal TS vias by a min-cost network flow heuristic, which will be presented in the next subsection. After the signal TS via planning, the inter-tier nets are decomposed into a set of 2D nets by adding pseudo-pins on each tier to replace the signal TS vias, and the decomposed nets are routed by 2D routing algorithms, e.g., 2D maze routing, at the last step to complete the initial 3D global routing.

The initial 3D global routing stage can also be completed by simultaneous signal TS via planning and wire routing, either with concurrent approaches or sequential approaches. The work in Ref. 22) extends the hierarchical routing algorithm [6] as a concurrent approach to the 3D global routing problem. However, it assumes space between cell rows is provided for TS vias, and only limits the total TS via per row of cells without modeling the placed cells as routing blockages. As a sequential approach, the work in Ref. 15) applies 3D maze searching to conduct the simultaneous routing of TS vias and wires. The multipin nets are first converted to minimum spanning trees, and then these minimum spanning trees are converted to minimum Steiner trees by performing a point-to-path 3D maze searching. Steiner edges are created when the searching path touches the existing edges of the tree before the target point. The maze-searching algorithm finds the shortest paths, with awareness of obstacles that are capable of handling TS vias by properly setting the routing cost and routing resources along the z-direction for the 3D maze searching engine.

2.2.2 Signal TS Via Planning

The signal TS via planning can be used at the initial 3D global routing stage, or at the fine-level routing refinement in the multilevel scheme. Given a planning window PW divided into planning bins $\{b_j\}$, $j = 1, 2, \ldots, m$, with position (x_j, y_j) and capacity c_j, assign signal TS vias $\{v_i\}$, $i = 1, 2, \ldots, n$, into one of these bins of PW, so that the TS via number assigned to each bin b_j does not exceed its capacity c_j, and the wirelength is minimized.

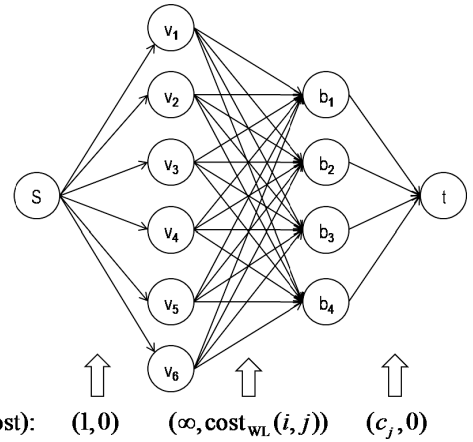


Fig. 7 Min-cost flow problem for signal TS via planning.

The min-cost network flow heuristic is a commonly used method for signal TS via planning [15),58)]. As shown in **Fig. 7**, a network $G(V, E)$ is constructed, whose node set includes all the TS vias $\{v_i\}$, all the planning bins $\{b_j\}$, a pseudo source node s, and a pseudo sink node t. There are three kinds of edges in the edge set E, where each edge will be assigned with a (*capacity, cost*) pair:

- The source node s has supply of n, and connects to n TS vias $\{v_i\}$. Each edge (s, v_i) has capacity 1 and cost 0.
- There are $n \times m$ edges from the TS vias $\{v_i\}$ to the bins $\{b_j\}$. The capacity of edge (v_i, b_j) is infinity, and the cost $\text{cost}_{WL}(i, j)$ is the estimated wirelength after assigning v_i to b_j.
- Every bin b_j connects to the sink node t, where each edge (b_j, t) has a capacity of c_j and a cost of 0.

In this min-cost flow problem, the supply at the source node and the maximal capacities at the edges are all integers. Thus, the optimal solution will also have integer values [30)]. This solution indicates the optimal assignment of each signal TS via to the planning bins, in the sense of the estimated wirelength, and can be solved in polynomial time.

2.2.3 Thermal TS Via Planning

Thermal TS via planning is carried out after signal TS via planning, because wirelength is usually more critical. During thermal TS via planning for an L-tier 3D IC, the routing region on each tier is divided into $N \times M$ planning bins, which are denoted as $\{b_{i,j,k}\}$ with $1 \le i \le N$, $1 \le j \le M$ and $1 \le k \le L$. Given the placement of cells/macros and the signal TS via planning results, we can compute the TS via capacity $c_{i,j,k}$ and the minimum TS via number $s_{i,j,k}$ of each planning bin $b_{i,j,k}$, which give the per-bin TS via constraints such that the TS via number $n_{i,j,k}$ satisfies $s_{i,j,k} \le n_{i,j,k} \le c_{i,j,k}$. Thus, the problem of thermal TS via planning is to minimize the total number of TS vias $\sum_{i,j,k} n_{i,j,k}$, subject to the temperature constraint and the per-bin TS via constraints.

The thermal TS via planning problem can be solved by linear programming [58] with $\{n_{i,j,k}\}$ as problem variables. However, the approximation that the temperature change $\Delta T_{i,j,k}$ in bin $b_{i,j,k}$ is proportional to the TS via number change $\Delta n_{i,j,k}$ is not accurate in general, as the thermal conductance and the temperature are in a nonlinear relationship. The thermal sensitivity factors depend on the TS via distribution. Therefore, the thermal sensitivity analysis and the linear programming have to be performed iteratively until a feasible solution is reached.

Thermal TS via insertion affects the thermal characteristic of a 3D IC. The work in Ref. 16) proposes a nonlinear programming-based method, where a thermal resistive network model [52] is integrated in the formulation. The bin structure is shown in **Fig. 8** (a), where each bin $b_{i,j,k}$ is associated with a power dissipation $P_{i,j,k}$ and a temperature $T_{i,j,k}$. The thermal resistive network model according to the bin structure is shown in Fig. 8 (b). It is assumed that the heat sink is attached to the bottom tier, and the other sides of the resistive network are adiabatic. The notations for the heat flows in the thermal resistive network are shown in Fig. 8 (c), where a heat flow opposite to the arrow direction is represented by a negative value. The power dissipation and the heat flows related to a node in the thermal resistive network satisfy:

$$P_{i,j,k} + H^{(x)}_{i-1,j,k} + H^{(y)}_{i,j-1,k} + H^{(z)}_{i,j,k+1} = H^{(x)}_{i,j,k} + H^{(y)}_{i,j,k} + H^{(z)}_{i,j,k} \tag{1}$$

And the heat flows and the temperature satisfy:

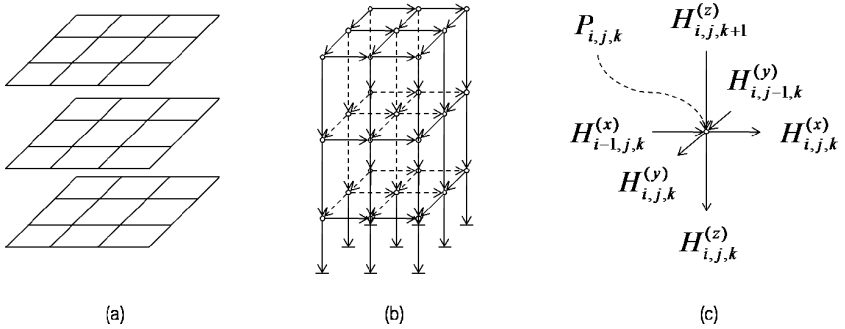


Fig. 8 TS via planning bins and the thermal resistive network model.

$$\begin{aligned} H^{(x)}_{i,j,k}/G^{(x)}_{i,j,k} &= (T_{i,j,k} - T_{i+1,j,k}) \\ H^{(y)}_{i,j,k}/G^{(y)}_{i,j,k} &= (T_{i,j,k} - T_{i,j+1,k}) \\ H^{(z)}_{i,j,k}/G^{(z)}_{i,j,k} &= (T_{i,j,k} - T_{i,j,k-1}) \end{aligned} \tag{2}$$

where $G^{(x)}_{i,j,k}$, $G^{(y)}_{i,j,k}$ and $G^{(z)}_{i,j,k}$ are the thermal conductivities at the heat flow edge $(b_{i,j,k}, b_{i+1,j,k})$, $(b_{i,j,k}, b_{i,j+1,k})$, $(b_{i,j,k}, b_{i,j,k-1})$, respectively.

Instead of directly formulating $\{n_{i,j,k}\}$ as problem variables, the temperatures $T = \{T_{i,j,k}\}$ and the heat flows $H = \left\{H^{(x)}_{i,j,k}\right\} \cup \left\{H^{(y)}_{i,j,k}\right\} \cup \left\{H^{(z)}_{i,j,k}\right\}$ are the variables in the nonlinear programming problem. The temperature and heat flows determine the number of TS vias $n_{i,j,k}(H,T)$ to be planned in each bin $b_{i,j,k}$:

$$\begin{aligned} & n_{i,j,k}(H,T) g_{TSV} + g^{(z)}_{i,j,k} = G^{(z)}_{i,j,k} = H^{(z)}_{i,j,k}/(T_{i,j,k} - T_{i,j,k-1}) \\ \Rightarrow & n_{i,j,k}(H,T) = \left(H^{(z)}_{i,j,k}/(T_{i,j,k} - T_{i,j,k-1}) - g^{(z)}_{i,j,k}\right)/g_{TSV} \end{aligned} \tag{3}$$

where $g^{(z)}_{i,j,k}$ is the thermal conductivity at the heat flow edge $(b_{i,j,k}, b_{i,j,k-1})$ without any TS vias, and g_{TSV} is the thermal conductivity of one TS via.

Thus, the thermal TS via planning problem can be solved by minimizing $n_{i,j,k}(H,T)$, subject to the per-bin TS via constraints $s_{i,j,k} \le n_{i,j,k}(H,T) \le c_{i,j,k}$, the temperature constraints $T_{i,j,k} \le T_{\max}$ and the thermal model constraints as in Eqs. (1) and (2). This nonlinear programming problem can be solved by the

alternating direction TS via planning algorithm (ADVP) [16], which iteratively alternates between vertical thermal TS via planning and horizontal thermal TS via planning.

3. 3D Placement

Placement is an important step in the 3D physical design flow. The performance, power, temperature and routability are significantly affected by the quality of placement results. Thus, a 3D placement tool has to minimize the total wirelength, and has to control the TS via number and temperature.

3.1 Problem Formulation

Given a circuit $H = (V, E)$, the tier number K and the per-tier placement region $R = [0, a] \times [0, b]$, where V is the set of cell instances (represented by vertices) and E is the set of nets (represented by hyperedges) in the circuit H (represented by a hypergraph), a placement (x_i, y_i, z_i) of the cell $v_i \in V$ satisfies that $(x_i, y_i) \in R$ and $z_i \in \{1, 2, \ldots, K\}$. The 3D placement problem is to find a placement (x_i, y_i, z_i) for every cell $v_i \in V$, so that an objective function such as the weighted total wirelength is minimized, subject to overlap-free constraints, and other constraints such as performance and temperature. In this paper we focus on temperature constraints, as the performance constraints are similar to that of 2D placement. The reader may refer to Refs. 20), 42) for a survey and tutorial of 2D placement.

3.1.1 Wirelength Objective Function

The quality of a placement solution can be measured by the performance, power and routability, but the measurement is more difficult than that in routing. In order to model these aspects during optimization, the weighted total wirelength is a widely accepted metric for measuring placement quality [41),42)]. Formally, the placement objective function is defined as

$$OBJ = \sum\nolimits_{e \in E} (1 + r_e) \cdot (WL(e) + \alpha_{TSV} \cdot TSV(e)) \tag{4}$$

The objective function depends on the placement $\{(x_i, y_i, z_i)\}$, and it is a weighted sum of the wirelength $WL(e)$ and the number of TS vias $TSV(e)$ over all the nets. The weight $(1 + r_e)$ reflects the criticality of the net e, which is usually related to performance optimization. The unweighted wirelength is represented by setting r_e to 0. This weight is often used to model thermal effect, timing and timing criticality of net e [26)].

The wirelength $WL(e)$ is usually estimated by the half-perimeter wirelength (HPWL) [19),26)]:

$$WL(e) = \left(\max_{v_i \in e}\{x_i\} - \min_{v_i \in e}\{x_i\}\right) + \left(\max_{v_i \in e}\{y_i\} - \min_{v_i \in e}\{y_i\}\right) \tag{5}$$

Similarly, $TSV(e)$ is modeled by the range of $\{z_i : v_i \in e\}$ [19),25),26)]:

$$TSV(e) = \max_{v_i \in e}\{z_i\} - \min_{v_i \in e}\{z_i\} \tag{6}$$

The coefficient a_{TSV} is the weight for TS vias; it models a TS via as some additional wirelength. For example, in 0.18 μm silicon-on-insulator (SOI) technology, Ref. 23) estimates that a 3 μm-thickness TS via is roughly equivalent to 8 to 20 μm of metal-2 wire in terms of capacitance, and it is equivalent to about 0.2 μm of metal-2 wire in terms of resistance. Thus a coefficient α_{TSV} between 8 to 20 (μm) can be used for optimizing power or delay in this case.

3.1.2 Overlap-Free Constraints

The ultimate goal of overlap-free constraints can be expressed as the following:

$$\begin{array}{l} |x_i - x_j| \geq (w_i + w_j)/2 \\ \qquad\qquad \text{or} \qquad\qquad \text{for all cell pairs } v_i, v_j \text{ with } z_i = z_j \\ |y_i - y_j| \geq (h_i + h_j)/2 \end{array} \tag{7}$$

where (x_i, y_i, z_i) is the placement of cell i, and w_i and h_i are its width and height, respectively. The same applies to cell j. Such constraints were used directly in some analytical placers early on, such as in Ref. 7).

However, this formulation leads to a large number of $O(n^2)$ either-or constraints, where n is the total number of cells. This amount of constraints is not practical for modern large-scale designs.

To formulate and handle these pairwise overlap-free constraints, modern placers use a more scalable procedure to divide the placement into *coarse legalization* and *detailed legalization*. Coarse legalization relaxes the pairwise non-overlap constraints by using regional density constraints:

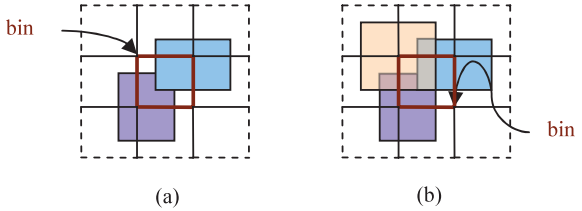


Fig. 9 (a) Density constraint is satisfied; (b) Density constraint is not satisfied [54].

$$\sum_{\substack{\text{for all } cell_i \\ \text{with } z_i=k}} overlap(bin_{m,n,k}, cell_i) \leq area(bin_{m,n,k}) \quad (\text{for all } m,n,k) \qquad (8)$$

where $overlap(bin_{m,n,k}, cell_i)$ represents the partial area of $cell_i$ that is contained in $bin_{m,n,k}$, and $area(bin_{m,n,k})$ represents the area capacity of $bin_{m,n,k}$.

For a 3D circuit with K tiers, each tier is divided into $L \times M$ bins. If every $bin_{l,m,k}$ satisfies inequality (8), the coarse legalization is finished. Examples of the density constraints on one tier are given in **Fig. 9**.

After coarse legalization, the detailed legalization is to satisfy pairwise non-overlap constraints, using various discrete methods and heuristics [19),26)].

3.1.3 Thermal Awareness

In existing literature, temperature issues are not directly formulated as constraints. Instead, a thermal penalty is appended to the wirelength objective function to control the temperature. This penalty can either be the weighted temperature penalty that is transformed to thermal-aware net weights [26)], or the thermal distribution cost penalty [56)], or the distance from the cell location to the heat sink during legalization [19)].

3.2 Overview of Existing 3D Placement Approaches

The state-of-the-art algorithms for 2D placement can be classified into flat placement approach, top-down partitioning-based approach, and multilevel placement approach [42)]. These approaches exhibit scalability for the growing complexity of modern VLSI circuits. In order to handle the scalability issues, these approaches divide the placement problem into three stages of global placement, legalization, and detailed placement. Given an initial solution, the global placement refines the solution until the overlap-free constraints (Section 3.1.2) are satisfied. These regions are handled in a top-down fashion from coarsest level to finest level by the partitioning-based techniques and the multilevel placement techniques, and are handled in a flat fashion at the finest level by the flat placement techniques. After the global placement, legalization proceeds to determine the specific location of all cells without overlaps, and the detailed placement performs local refinements to obtain the final solution.

As the modern 2D placement approaches evolve, a number of 3D placement approaches are also being developed to address the issues of 3D IC technologies. Most of the existing approaches, especially at the global placement stage, can be viewed as extensions of 2D placement approaches. We group the 3D placement approaches into four categories: partitioning-based approach, flat placement approaches, multilevel placement approach, and transformation-based approach.

- The partitioning-based approach [1),2),21),26)] applies the same divide-and-conquer strategy as the well-known partitioning-based 2D placement approach. The bisection of the placement region in the z-direction is performed at some suitable steps in addition to the bisections in the x-direction and the y-direction. And the cost of partitioning is measured by a weighted sum of the estimated wirelength and the TS via number, where the nets can be further weighted by thermal-aware or congestion-aware factors to consider temperature and routability.
- Flat placement approaches are the variations of quadratic placement, including the force-directed approach [25),31)], the cell-shifting approach [29)], and the quadratic uniformity modeling approach [56)]. Since the unconstrained quadratic placement will introduce a great amount of cell overlaps, different variations are developed for overlap removal. The minimization of the quadratic wirelength, as well as the quadratic form of TS via number, could be transformed to the problem of solving a linear system. The idea of these flat placement approaches is to append a *force vector*, which is computed from the area density distribution and helps to remove overlaps, to the right-hand side of the linear system. The vector is updated and the linear system is solved iteratively until the area in every pre-defined region is not greater than the area capacity of that region. These flat placement approaches differ

by the definition of this force vector, which will be presented in detail in Section 3.3.

- The multilevel placement approach [13] constructs a physical hierarchy from the original netlist, and solves a sequence of placement problems from the coarsest level to the finest level. An analytical 3D placement solver is applied at each level, which optimizes the log-sum-exp wirelength [5),43)] and the log-sum-exp TS via number estimation subject to the overlap-free constraints. To model the 3D overlap-free constraints for the intermediate solution, which is continuous at the z-direction, the area projection method with pseudo tiers is applied to guarantee the legality of the final solution. Details will be presented in Section 3.3.
- In addition to these approaches, the 3D placement approach proposed in Ref. 19) makes use of existing 2D placement results and constructs a 3D placement by transformation. The transformation schemes include two folding transformations, the stacking transformation and the window-based folding/stacking transformation. All these transformations start with a wirelength-optimized 2D placement on a placement region, whose width and height are $\sqrt{K}$ times as large as the width and height of a K-tier 3D IC. The idea of *folding transformations* is to fold the 2D placement like a piece of paper without cutting off any parts of the placement. Thus, the lengths of the global nets that go across the folding lines get reduced. The *stacking transformation* first shrinks the 2D placement by a factor of $\sqrt{K}$, which can be viewed as a wirelength-optimized 3D placement projected on the (x, y) plane. Then the Tetris-style legalization is used to decide the tier assignment of the stacked cells. Although the wirelength is small by the stacking transformation, the TS via number is usually large. To trade off the wirelength and TS via number, a *window-based folding/stacking transformation* can be used, which divides the 2D placement into windows and transforms each window to 3D placement by the folding transformation or the stacking transformation.

3.3 Modeling of 3D Overlap-Free Constraints

3D global placement by the flat placement approaches and the multilevel placement approach, as presented in Section 3.2, usually relax the tier assignment from the discrete set $z \in \{1, 2, \ldots, K\}$ to a continuous interval $z \in [1, K]$ for a K-tier 3D IC. The modeling of 3D overlap-free constraints for such intermediate placement solution is an essential issue for these 3D placement approaches.

The flat placement approaches (force-directed, cell-shifting, and quadratic uniformity modeling) define the cell area distribution in the 3D space in the following way: for a K-tier 3D IC with width W and height H, a 3D space $[0, W] \times [0, H] \times [0, \tau K]$ is defined; a cell with width w and height h and its lower left corner at (x, y, z) occupies the 3D region $[x, x + w] \times [y, y + h] \times [z - \tau, z]$. In such a way, the cell area distribution in the 3D space is defined for a given intermediate placement solution.

The force-directed approach [25),31)] computes the force vector (see Section 3.2) by solving a 3D Poisson equation for the potential of the cell area distribution. The force vector is the gradient of the 3D potential field. The cell-shifting approach [29] first computes the expected placement by cell-shifting to even out the cell area distribution; this expected placement is not actually performed, and a pseudo net is created for each cell, where the pseudo pins are located properly so that the cells tend to move in the desired direction; the steepest descent direction of the wirelength for these pseudo nets gives the force vector in the linear system. The quadratic uniformity modeling approach [56] defines a density penalty function based on the 3D discrete cosine transformation (DCT) of the cell area distribution, and approximates this density penalty function by a quadratic function. The steepest descent direction of the density penalty function is the force vector appended to the right-hand side of the linear system for this approach.

The multilevel placement approach [13] models the 3D overlap-free constraints in a different way. Its analytical engine solves the 3D global placement problem as a nonlinear programming problem. The tier assignment is also explored in the interval $z \in [1, K]$. Instead of defining a cell area distribution in the 3D space, this analytical engine models the overlap-free constraints by examining the area distribution on certain cross sections in the 3D space. The cross sections include all the actual tiers and all pseudo tiers between every two adjacent tiers, where $z \in \{1, 2, \ldots, K\}$ and $z \in \{3/2, 5/2, \ldots, (2K - 1)/2$, respectively. The area distribution on a specific actual tier or a pseudo tier is defined by an area projection function based on the bell-shaped function [43]. For a 3D placement

problem without white space, which can be achieved by adding dummy cells, it can be proved that if the area distributions on all the actual tiers and pseudo tiers are equal, the placement will be legal. These area distribution constraints imply the 3D overlap-free constraints. Thus, the 3D global placement is formulated as a nonlinear programming problem:

$$\begin{array}{lll} \text{minimize} & WL(x,y,z) + \alpha \cdot TSV(x,y,z) & \\ \text{subject to} & D_k(u,v) = 1 & \text{for } k = 1,2,\ldots,K \\ & D_k(u,v) = 1 & \text{for } k = \dfrac{3}{2}, \dfrac{5}{2}, \ldots, \dfrac{2K-1}{2} \end{array} \quad (9)$$

where (x_i, y_i) is the placement of cell v_i assigned to tier z_i, and the density function $D_k(u,v)$ is the sum of the area contribution of all the cells to point (u,v) at an actual tier k or a pseudo tier k. The cell area density function $d_i(u,v)$ is 1 inside the region covered by v_i, and it is 0 outside this region. The area contribution is computed after area projection $\eta(k,z)$. These functions are defined as:

$$D_k(u,v) = \sum_i \eta(k,z_i) d_i(u,v)$$

$$\eta(k,z) = \begin{cases} 1 - 2(z-k)^2 & |z-k| \le 1/2 \\ 2(|z-k|-1)^2 & 1/2 < |z-k| \le 1 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

The density functions $D_k(u,v)$ can be converted to differentiable functions by the density smoothing techniques, e.g., Helmholtz smoothing [12]. Thus, the nonlinear programming problem can be solved by the quadratic penalty method, or the augmented Lagrangian method, to obtain a 3D global placement solution.

4. 3D Floorplanning

3D IC technologies make floorplanning a much more difficult problem because the multi-tier structures dramatically enlarge the solution space and the increased power density accentuates the thermal problem. Therefore, moving to 3D designs increases the problem complexity greatly:

- The design space of 3D floorplanning increases exponentially with the number of tiers. The work in Ref. 36) showed that, given a floorplanning problem with n blocks, the solution space of 3D floorplanning with L tiers is $n^{L-1}/(L-1)!$ times larger than the solution space of 2D floorplanning, if a 3D floorplan solution is represented by an array of the corresponding 2D floorplan representations.
- The addition of a temperature constraint or temperature minimization objective complicates optimization, requiring tradeoffs among area, wirelength, and thermal characteristics. And with the high temperature in 3D chips, it is necessary to account for the closed temperature/leakage power feedback loop to accurately estimate or optimize either one.
- Multi-tier stacking offers a reduction in inter-block latency. It can also be used to help the intra-block wire latency when the block is implemented in multiple tiers. Use of multi-tier blocks requires a novel physical design infrastructure to explore 3D design space.

Therefore, it is imperative to develop thermal-aware and timing-aware floorplanning tools that consider 3D design constraints. The goal of 3D floorplanning is to pack blocks on multiple tiers with no overlaps by optimizing some objectives without violating some design constraints. According to the block representation, we can classify the 3D floorplanning problem into two types. The first type is a 3D floorplan with 2D blocks in which each block is a 2D rectangle and the packing on each tier can be treated as a 2D floorplan. A 3D floorplan with 2D blocks can be represented by an array of 2D representations (2D array), each representing all blocks located on one tier. The second type of 3D floorplanning involves 3D blocks where each block is treated as a cuboid block with non-zero height in the z-dimension. In this case, the existing 2D representations no longer apply, and we need new representations.

One important application of 3D floorplanning is to provide 3D physical prototyping for microarchitectural evaluation. Recent studies have provided block models for various microarchitectural structures, including 3D cache [33),47),50)], 3D register files [49)], 3D arithmetic units [45)], and 3D instruction scheduler [46)]. Therefore, to utilize 3D blocks, the decision cannot simply be made from the architecture side only or the physical design side only. To enable the co-optimization between 3D microarchitectural and physical design, a true 3D packing engine is needed to choose the implementation while performing the packing optimization.

Due to the page limitations, using 3D floorplanning for 3D microarchitectural exploration is not covered in this paper, and the readers may refer to Refs. 28), 37), 53) for details.

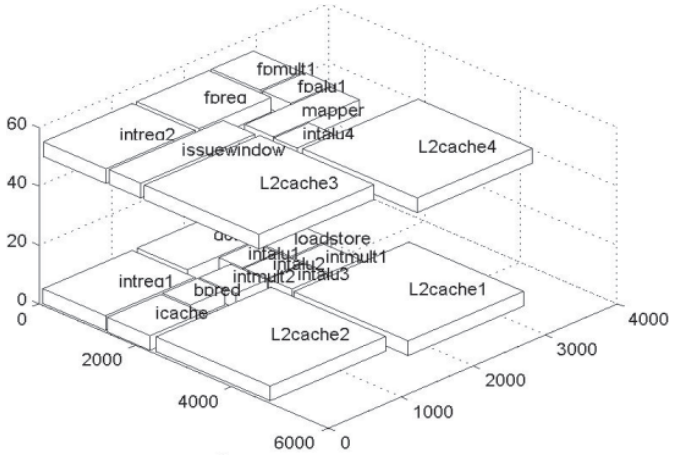
4.1 Problem Formulation

Similar to the traditional 2D floorplanning, 3D floorplanning also aims at a small packing area, short wirelength, low power consumption and high performance. Although 3D IC technologies have many potential benefits, thermal distribution becomes a critical issue during every step of 3D physical design. Therefore, 3D floorplanning distributes blocks on a certain number of tiers without overlapping each other so that the design metrics, such as the chip area, wirelength, TS via number and maximal on-chip temperature, are optimized or meet some design constraints.

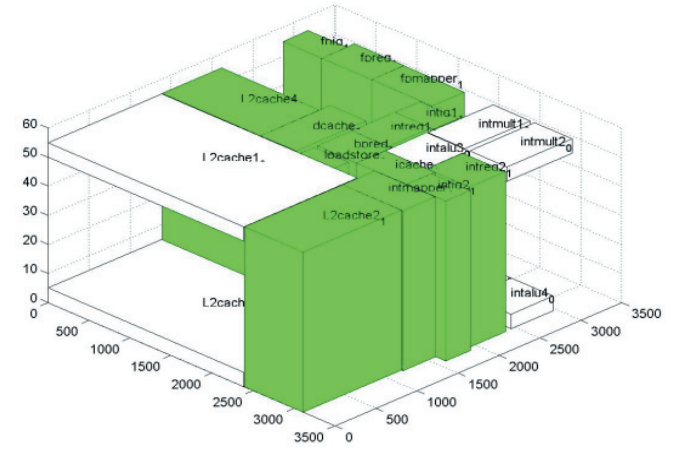
With the additional z-direction, not only can the 2D blocks be spread among multiple tiers, but some individual components can be folded into the designs of a multi-tier block so that the intra-block wire latency can be reduced, as well as the power consumption. The 3D components with different tier numbers can be treated as cuboid blocks to be packed in the 3D space. The dimension in the z-direction represents the tier information. Therefore, in 3D floorplanning the blocks to be packed can be 2D blocks or 3D blocks. **Fig. 10** (a) shows the 2-tier packing for Alpha 21264 in which all blocks are 2D blocks, and Fig. 10 (b) shows the packing with some 3D blocks. The implementation for each 3D component may have multiple choices with different area-delay-power tradeoffs. As shown in Fig. 10 (b), it is possible that an optimal floorplan has a subset of the microarchitectural units occupying a single tier, while others are implemented on multiple tiers with potentially different heights in the z-dimension. According to the block representation, we classify the 3D floorplanning problem into two types: 3D floorplan with 2D blocks only and 3D floorplan with possible 3D blocks.

4.1.1 3D Floorplanning with 2D Blocks

Though 3D packing with 2D blocks can be treated as multiple stacked 2D packings, the additional concern at the chip level relates to the large number of active devices that are packed into a much smaller area, so that the power density is much higher than in a corresponding 2D circuit. As a result, in addition to the common objectives of packing area and wirelength, thermal issues are given



(a) 3D floorplanning with 2D blocks



(b) 3D floorplanning with 3D blocks

Fig. 10 3D floorplanning [54].

primacy among the set of design objectives. Hence, we can formulate a 3D floorplan with 2D blocks as follows.

An instance of the 3D floorplanning problem with 2D blocks is composed of a

set of blocks $\{m_1, m_2, \cdots m_n\}$. A block m_i is a $W_i \times H_i$ rectangle with area A_i, aspect ratio H_i/W_i, and power density PD_i. Each block is free to rotate. There is a fixed number of tiers L. Let the tuple $(x_i,\ y_i,\ l_i)$ denote the coordinates of the bottom-left corner of block m_i, where $1 \leq l_i \leq L$. A 3D floorplan F is an assignment of $(x_i,\ y_i,\ l_i)$ for each block m_i such that no two blocks overlap. The common objectives of 3D floorplanning algorithms are to minimize (1) chip peak temperature $T_{\max}$, (2) total wirelength (or total power), and (3) chip area. Chip area is the product of the maximum height and width over all tiers. Wirelength is the half perimeter wirelength estimation. In addition, some other design objectives, such as noise, performance, the number of TS vias, etc., can be considered at the same time. Also, some design constraints can be included, such as pre-packed blocks (the positions of the constrained blocks are pre-defined), alignment constraints (some specific blocks are constrained to be aligned in x, y or z directions), etc.

Since 3D floorplanning with 2D blocks can be represented with an array of 2D representations, the 2D floorplanning algorithm can be extended to handle multi-tier designs by introducing new operations in optimization techniques. Though floorplanning for 2D design is a well-studied problem, with the extension of an IC at the z-direction, the design space of 3D IC floorplanning increases exponentially. Though the multi-tier design can be represented by an array of 2D packings, the specific optimization techniques are still needed for efficient exploration. Thermal-aware optimization is especially critical in 3D designs.

4.1.2 3D Floorplanning with 3D Blocks

Fine-grain 3D IC provides reduced intra-block wire delay as well as improved power consumption. The implementation for each component may have multiple choices due to various configurations. Therefore, the components might be implemented on multiple tiers, such as a 4-tier or 2-tier cache, by different stacking techniques. But locally, the best implementation of an individual unit may not necessarily lead to the best design for the entire multi-tiered chip. To obtain the trade-off between multiple objectives, it is possible to have cubic blocks, which have different heights in the z-direction, in the packing design. Therefore, a cube-packing algorithm should be developed to arrange the given circuit components in a rectangular box of the minimum volume without overlapping each other. With the various implementations for each critical component, the block implementation is partially defined. Without the physical information, it is impossible to obtain the optimal implementations for components for the final chip. Thus, 3D floorplanning with 3D blocks should not only determine the coordinates of the blocks, but also be able to choose the configurations for components, such as the number of tiers, the partitioning approaches, etc. Therefore, we can formulate the 3D packing with 3D blocks as follows.

Given a list of 3D blocks: Suppose for block i, there are k different implementations that are recorded in a candidate list as $\{c_1^i, c_2^i, \ldots c_k^i\}$. And each candidate c_j^i has the width w_j^i, height h_j^i, tier number z_j^i, delay d_j^i, and power p_j^i (assume each tier has the same power consumption). The objective is to generate a floorplan that optimizes for the die area, maximum on-chip temperature, etc. At the same time, the number of tiers is normally fixed, which means, given the tier number constraints as Z_{con}, the blocks should not exceed the tier number constraint.

4.2 3D Floorplanning Algorithms

Since the 2D and 3D rectangular packing problems are NP-hard, most floorplanning algorithms are based on stochastic combinatorial optimization techniques such as simulated annealing [32] and genetic algorithm [40]. But analytical algorithms [59] are also proposed for handling 3D floorplanning. In this section, we focus on the simulated annealing algorithm, which is to minimize a given cost function by searching the solution space represented by a specific representation. Normally, the cost function describes the combination of chip area, wirelength, maximal on-chip temperature, or other factors.

Figure 11 shows the optimization flow based on the simulated annealing approach. The critical components in a simulated annealing algorithm include: (1) cooling schedule, (2) cost function, (3) representation of the solution, (4) solution perturbation. The whole cooling schedule includes the set up of the initial temperature, cooling function, and end temperature, all of which depend on the size of the problem and the property of the problem. The cost function is usually a weighted sum of the wirelength estimate (half-perimeter model), the total area of all tiers (product of the maximal height and width and number of tiers), the number of TS vias and the maximal temperature. Various 3D floorplanning

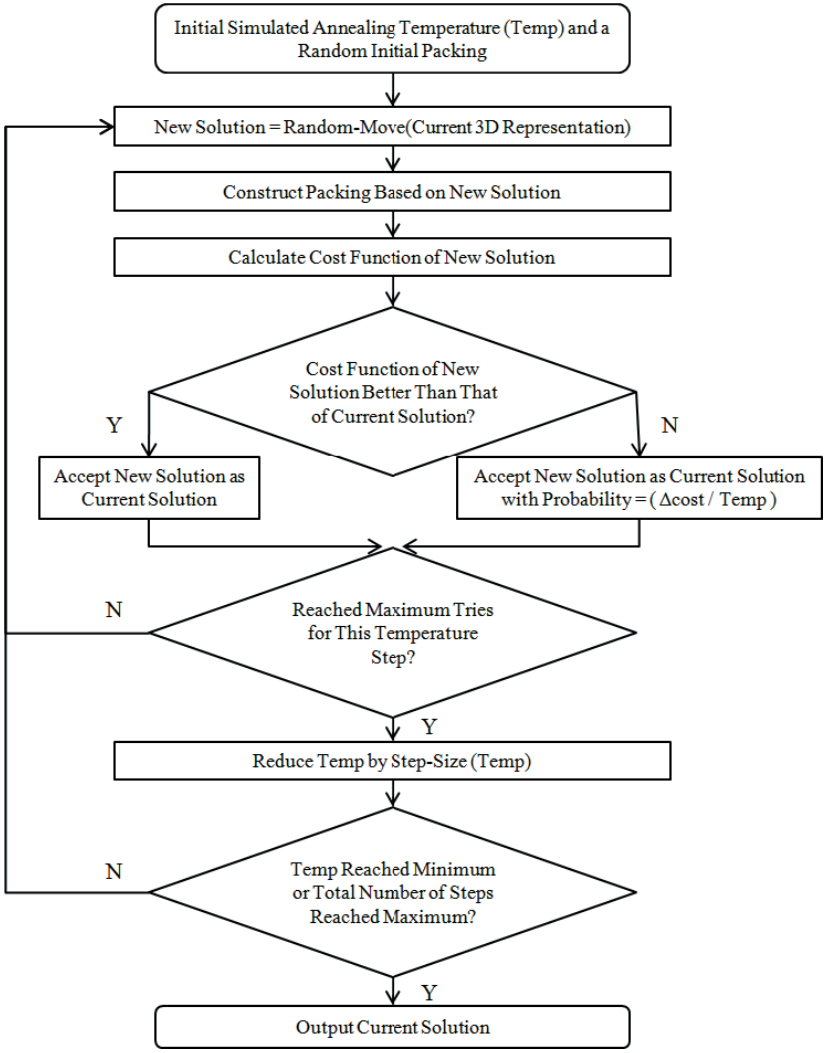


Fig. 11 The flow of the simulated annealing approach [37].

Table 1 Various representations for 2D floorplanning [54].

Representation	Solution space	Complexity of floorplan construction	Move	Packing category
NPE (SST)	$O(n!2^{3n-3}/n^{1.5})$	$O(n)$	O(1)	Slicing
SP	$n!^2$	$O(n\log\log n) - O(n^2)$	O(1)	General
BSG	$n!C(n^2,n)$	$O(n^2)$	O(1)	General
O-tree	$O(n!2^{2n}/n^{1.5})$	$O(n)$	O(1)	Compact
B*-tree	$O(n!2^{2n}/n^{1.5})$	$O(n)$	O(1)	Compact
CBL	$O(n!2^{3n-3}/n^{1.5})$	$O(n)$	O(1)	Mosaic
TCG	$n!^2$	$O(n^2)$	$O(n)$	General

Table 2 Various representations for 3D floorplanning with 3D blocks [54].

Representation	Complexity of floorplan construction	Move complexity	Packing category	Solution space
ST [55]	$O(n^2)$	O(1)	General but not all	$n!^3$
Squin [55]	$O(n^2)$	O(1)	all	$n!^5$
3D Slicing-tree [8]	$O(n)$	O(1)	slicing	$O(n!3^{n-1}2^{2n-2}/n^{1.5})$
3D-subTCG [57]	$O(n^2)$	$O(n^2)$	General but not all	$n!^3$
3D-CBL [38]	$O(n)$	O(1)	Mosaic	$O(n!\ 3^{n-1}2^{4n-4})$

algorithms differ in the solution representation, which defines the neighborhood structure for solution permutation.

The 3D floorplan with 2D blocks can be represented as an array of 2D floorplans at each tier; thus, the solution of 3D floorplanning with 2D blocks can be represented as an array of the solution representation of the corresponding 2D floorplans. There is a plethora of literature on 2D floorplanning problems, so we only summarize the various representations in **Table 1**. The solution perturbation includes:

Rotation, which rotates a block;

Swap, which swaps two blocks on one tier;

Reverse, which exchanges the relative position of two blocks on one tier;

Move, which moves a block from one side (such as top) of a block to another side (such as left);

Inter-tier swap, which swaps two blocks at different tiers;

z-neighbor swap, which swaps two blocks at different tiers but close to each other;
z-neighbor move, which moves a block to a position at another tier close to the current position.

The 3D floorplanning with 3D blocks also has various solution representations, which are summarized in **Table 2**. The solution representations also define the neighborhood of solution perturbation in the simulated annealing algorithm. The readers may refer to the references in Table 2 for more details.

5. Conclusions

Along with the development of 3D IC technologies in the recent decade, a significant amount of advancement has been made in the 3D IC physical design automation. In this paper we cover important problems and algorithms developed in the 3D IC physical design flow, including 3D routing, thermal TS via planning, 3D placement, and 3D floorplanning. A high-level overview of the basic concepts in the 3D physical design flow is presented, and the necessary references are included for readers who would like to dig deeper in a specific topic.

To facilitate wide adoption of 3D IC design technology, future 3D physical design research needs to address the following issues:

1) 3D physical hierarchy optimization: The study in Ref. 11) highlighted the difference of physical hierarchy and logic hierarchy, and underscored the need of physical design generation. This is even more important for 3D designs. Early 3D architecture exploration work using block stacking only led to disappointing performance gain due to simple adoption of logic and physical hierarchy optimized for 2D designs [18]. More performance gain is shown to be possible on the same architecture by applying 3D designs further down to the logic hierarchy [37].
2) 3D mixed-size placement under TSV density constraint: In order to effectively explore physical hierarchy, we need highly scalable 3D mixed-size placement algorithms to be applied to the flattened logic hierarchy as in the case for 2D designs [11]. In particular, it needs to handle the TS via density constraints, both globally per device layer and locally for every region in each device layer. The existing 3D placement algorithms lack capability to handle mixed-size placement and local TS via density support.
3) Commercial support for 3D physical design has been lacking. It will be very useful for the physical design research community to work together to develop a complete 3D physical design flow with interfaces to the existing commercial EDA tools for 2D design to facilitate the adoption of 3D IC design technologies. A promising integration framework is the OpenAccess 3D extension presented in Ref. 14) and this paper.
4) Strong linkage between the architecture level analysis tool and 3D physical planning tools is required to take advantage of 3D IC technologies with new architectures and physical implementations. Physical design and microarchitecture co-design is needed.

Acknowledgments This study is supported by the National Science Foundation (NSF) under CCF-0430077 and CCF-0528583.

References

1) Ababei, C., Mogal, H. and Bazargan, K.: Three-Dimensional Place and Route for FPGAs, *Proc. 2005 Conference on Asia South Pacific Design Automation*, pp.773–778 (2005).
2) Balakrishnan, K., Nanda, V., Easwar, S. and Lim, S.K.: Wire Congestion and Thermal Aware 3D Global Placement, *Proc. 2005 Conference on Asia South Pacific Design Automation*, pp.1131–1134 (2005).
3) Banerjee, K., Souri, S.J., Kapur, P. and Saraswat, K.C.: 3-D ICs: A Novel Chip Design for Improving Deep-Submicrometer Interconnect Performance and Systems-on-Chip Integration, *Proc. IEEE*, Vol.89, No.5, pp.602–633 (2001).
4) Bernstein, K., Andry, P., Cann, J., Emma, P., Greenberg, D., Haensch, W., Ignatowski, M., Koester, S., Magerlein, J., Puri, R. and Young, A.: Interconnects in the Third Dimension: Design Challenges for 3D ICs, *Proc. 44th Annual Conference on Design Automation*, pp.562–567 (2007).
5) Bertsekas, D.P.: Approximation Procedures Based on the Method of Multipliers, *Journal of Optimization Theory and Applications*, Vol.23, No.4, pp.487–510 (1977).
6) Burstein, M. and Pelavin, R.: Hierarchical Wire Routing, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, Vol.2, No.4, pp.223–234 (1983).
7) Chan, T.F., Cong, J., Kong, T. and Shinnerl, J.R.: Multilevel Optimization for Large-Scale Circuit Placement, *Proc. 2000 IEEE/ACM International Conference on Computer-aided Design*, pp.171–176 (2000).
8) Cheng, L., Deng, L. and Wong, M.D.F.: Floorplanning for 3-D VLSI Design, *Proc. 2005 Conference on Asia South Pacific Design Automation*, pp.405–411 (2005).

9) Chiang, C. and Sinha, S.: The Road to 3D EDA Tool Readiness, *Proc. 2009 Conference on Asia and South Pacific Design Automation*, pp.429–436 (2009).
10) Chiang, T.Y., Banerjee, K. and Saraswat, K.C.: Compact Modeling and SPICE-Based Simulation for Electrothermal Analysis of Multilevel ULSI Interconnects, *Proc. 2001 IEEE/ACM International Conference on Computer-Aided Design*, pp.165–172 (2001).
11) Cong, J.: Timing Closure Based on Physical Hierarchy, *Proc. 2002 International Symposium on Physical Design*, pp.170–174 (2002).
12) Cong, J., Luo, G. and Radke, E.: Highly Efficient Gradient Computation for Density-Constrained Analytical Placement, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, Vol.27, No.12, pp.2133–2144 (2008).
13) Cong, J. and Luo, G.: A Multilevel Analytical Placement for 3D ICs, *Proc. 2009 Conference on Asia and South Pacific Design Automation*, pp.361–366 (2009).
14) Cong, J. and Luo, G.: A 3D Physical Design Flow Based on OpenAccess, *Proc. International Conference on Communications, Circuits and Systems*, pp.1103–1107 (2009).
15) Cong, J. and Zhang, Y.: Thermal-Driven Multilevel Routing for 3-D ICs, *Proc. 2005 Conference on Asia South Pacific Design Automation*, pp.121–126 (2005).
16) Cong, J. and Zhang, Y.: Thermal Via Planning for 3-D ICs, *Proc. 2005 IEEE/ACM International Conference on Computer-Aided Design*, pp.745–752 (2005).
17) Cong, J. and Zhang, Y.: Thermal-Aware Physical Design Flow for 3-D ICs, *Proc. 23rd International VLSI Multilevel Interconnection Conference*, pp.73–80 (2006).
18) Cong, J., Jagannathan, A., Ma, Y., Reinman, G., Wei, J. and Zhang, Y.: An Automated Design Flow for 3D Microarchitecture Evaluation, *Proc. 2006 Asia and South Pacific Design Automation Conference*, pp.384–389 (2006).
19) Cong, J., Luo, G., Wei, J. and Zhang, Y.: Thermal-Aware 3D IC Placement Via Transformation, *Proc. 2007 Conference on Asia South Pacific Design Automation*, pp.780–785 (2007).
20) Cong, J., Shinnerl, J.R., Xie, M., Kong, T. and Yuan, X.: Large-scale Circuit Placement, *ACM Trans. Design Automation Electronic Systems*, Vol.10, No.2, pp.389–430 (2005).
21) Das, S.: *Design Automation and Analysis of Three-Dimensional Integrated Circuits*, Ph.D. Dissertation, Massachusetts Institute of Technology, Cambridge, MA (2004).
22) Das, S., Chandrakasan, A. and Reif, R.: Design Tools for 3-D Integrated Circuits, *Proc. 2003 Conference on Asia South Pacific Design Automation*, pp.53–56 (2003).
23) Davis, W.R., Wilson, J., Mick, S., Xu, J., Hua, H., Mineo, C., Sule, A.M., Steer, M. and Franzon, P.D.: Demystifying 3D ICs: The Pros and Cons of Going Vertical, *IEEE Design & Test of Computers*, Vol.22, No.6, pp.498–510 (2005).
24) Enbody, R.J., Lynn, G. and Tan, K.H.: Routing the 3-D Chip, *Proc. 28th Conference on ACM/IEEE Design Automation*, pp.132–137 (1991).
25) Goplen, B. and Sapatnekar, S.: Efficient Thermal Placement of Standard Cells in 3D ICs using a Force Directed Approach, *Proc. 2003 IEEE/ACM International Conference on Computer-Aided Design*, pp.86–89 (2003).
26) Goplen, B. and Sapatnekar, S.: Placement of 3D ICs with Thermal and Interlayer Via Considerations, *Proc. 44th Annual Conference on Design Automation*, pp.626–631 (2007).
27) Grove, A.S.: Physics and Technology of Semiconductor Devices, John Wiley & Sons, Inc., Hoboken, NJ (1967).
28) Healy, M., Vittes, M., Ekpanyapong, M., Ballapuram, C.S., Lim, S.K., Lee, H.H.S. and Loh, G.H.: Multiobjective Microarchitectural Floorplanning for 2-D and 3-D ICs, *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, Vol.26, No.1, pp.38–52 (2007).
29) Hentschke, R., Flach, G., Pinto, F. and Reis, R.: 3D-Vias Aware Quadratic Placement for 3D VLSI Circuits, *IEEE Computer Society Annual Symposium on VLSI*, pp.67–72 (2007).
30) Hillier, F.S. and Lieberman, G.J.: *Introduction to Operations Research 8th ed.*, McGraw-Hill Companies, Inc., New York (2004).
31) Kaya, I., Salewski, S., Olbrich, M. and Barke, E.: Wirelength Reduction Using 3-D Physical Design, *Proc. 14th International Workshop on Power and Timing Optimization and Simulation*, pp.453–462 (2004).
32) Kirkpatrick, S., Gelatt, C.D. and Vecchi, M.P.: Optimization by Simulated Annealing, *Science*, Vol.220, No.4598, pp.671–680 (1983).
33) Kleiner, M.B., Kuhn, S.A. and Weber, W.: Performance Improvement of the Memory Hierarchy of RISC-Systems by Application of 3-D-Technology, *Proc. 45th Electronic Components and Technology Conference*, pp.645–655 (1995).
34) Lee, S., Lemczyk, T.F. and Yovanovich, M.M.: Analysis of Thermal Vias in High Density Interconnect Technology, *Proc. 8th Annual IEEE Semiconductor Thermal Measurement and Management Symposium*, pp.55–61 (1992).
35) Li, X., Ma, Y., Hong, X., Dong, S. and Cong, J.: LP Based White Space Redistribution for Thermal Via Planning and Performance Optimization in 3D ICs, *Proc. 2008 Conference on Asia and South Pacific Design Automation*, pp.209–212 (2008).
36) Li, Z., Hong, X., Zhou, Q., Cai, Y., Bian, J., Yang, H.H., Pitchumani, V. and Cheng, C.K.: Hierarchical 3-D Floorplanning Algorithm for Wirelength Optimization, *IEEE Trans. Circuits and Systems I: Regular Papers*, Vol.53, No.12, pp.2637–2646 (2006).
37) Liu, Y., Ma, Y., Kursun, E., Reinman, G. and Cong, J.: Fine Grain 3D Integration for Microarchitecture Design Through Cube Packing Exploration, *The 25th International Conference on Computer Design*, pp.259–266 (2007).
38) Ma, Y., Hong, X., Dong, S. and Cheng, C.K.: 3D CBL: An Efficient Algorithm for General 3D Packing Problems, *Proc. 48th Midwest Symposium on Circuits and Systems*, Vol.2, pp.1079–1082 (2005).

39) Minz, J., Zhao, X. and Lim, S.K.: Buffered Clock Tree Synthesis for 3D ICs Under Thermal Variations, *Proc. 2008 Conference on Asia and South Pacific Design Automation*, pp.504–509 (2008).
40) Mitchell, M.: *An Introduction to Genetic Algorithms*, The MIT Press, Cambridge, MA (1998).
41) Nam, G.-J.: ISPD 2006 Placement Contest: Benchmark Suite and Results, *Proc. 2006 International Symposium on Physical Design*, pp.167–167 (2006).
42) Nam, G.-J. and Cong, J. (Eds.): *Modern Circuit Placement: Best Practices and Results*, Springer, New York, NY (2007).
43) Naylor, W.C., Donelly, R. and Sha, L.: Non-linear Optimization System and Method for Wire Length and Delay Optimization for an Automatic Electric Circuit Placer, US Patent 6301693, (Oct. 2001).
44) Pavlidis, V.F., Savidis, I. and Friedman, E.G.: Clock Distribution Networks for 3-D Integrated Circuits, *Proc. IEEE Custom Integrated Circuits Conference*, pp.651–654 (2008).
45) Puttaswamy, K. and Loh, G.H.: The Impact of 3-Dimensional Integration on the Design of Arithmetic Units, *Proc. 2006 IEEE International Symposium on Circuits and Systems*, pp.191–194 (2006).
46) Puttaswamy, K. and Loh, G.H.: Dynamic Instruction Schedulers in a 3-Dimensional Integration Technology, *Proc. 16th ACM Great Lakes symposium on VLSI*, pp.153–158 (2006).
47) Ronen, R., Mendelson, A., Lai, K., Shih-Lien, L., Pollack, F. and Shen, J.P.: Coming Challenges in Microarchitecture and Architecture, *Proc. IEEE*, Vol.89, No.3, pp.325–340 (2001).
48) Tong, C.C. and Wu, C.-L.: Routing in a Three-Dimensional Chip, *IEEE Trans. Comput.*, Vol.44, No.1, pp.106–117 (1995).
49) Tremblay, M., Joy, B. and Shin, K.: A Three Dimensional Register File for Superscalar Processors, *Proc. 28th Hawaii International Conference on System Sciences*, p.191 (1995).
50) Tsai, Y.-F., Xie, Y., Vijaykrishnan, N. and Irwin, M.J.: Three-Dimensional Cache Design Exploration Using 3DCacti, *Proc. 2005 IEEE International Conference on Computer Design: VLSI in Computers and Processors*, pp.519–524 (2005).
51) Westra, J., Bartels, C. and Groeneveld, P.: Probabilistic Congestion Prediction, *Proc. 2004 International Symposium on Physical Design*, pp.204–209 (2004).
52) Wilkerson, P., Furmanczyk, M. and Turowski, M.: Compact Thermal Modeling Analysis for 3D Integrated Circuits, *11th International Conference Mixed Design of Integrated Circuits and Systems*, pp.24–26 (2004).
53) Xie, Y., Loh, G.H., Black, B. and Bernstein, K.: Design Space Exploration for 3D Architectures, *J. Emerg. Technol. Comput. Syst.*, Vol.2, No.2, pp.65–103 (2006).
54) Xie, Y., Cong, J. and Sapatnekar, S.: *Three Dimensional Integrated Circuits Design: EDA, Design, and Microarchitectures*, Springer, New York (2009).
55) Yamazaki, H., Sakanushi, K., Nakatake, S. and Kajitani, Y.: The 3D-Packing by Meta Data Structure and Packing Heuristics, *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, Vol.83, No.4, pp.639–645 (2000).
56) Yan, H., Zhou, Q. and Hong, X.: Thermal Aware Placement in 3D ICs Using Quadratic Uniformity Modeling Approach, *Integration, the VLSI Journal*, Vol.42, No.2, pp.175–180 (2009).
57) Yuh, P.-H., Yang, C.-L. and Chang, Y.-W.: Temporal Floorplanning Using the Three-Dimensional Transitive Closure SubGraph, *ACM Trans. Design Automation of Electronic Systems*, Vol.12, No.4, pp.37 (2007).
58) Zhang, T., Zhan, Y. and Sapatnekar, S.S.: Temperature-Aware Routing in 3D ICs, *Proc. 2006 Conference on Asia South Pacific Design Automation*, pp.309–314 (2006).
59) Zhou, P., Ma, Y., Li, Z., Dick, R.P., Shang, L., Zhou, H., Hong, X. and Zhou, Q.: 3D-STAF: Scalable Temperature and Leakage Aware Floorplanning for Three-Dimensional Integrated Circuits, *Proc. 2007 IEEE/ACM International Conference on Computer-Aided Design*, pp.590–597 (2007).
60) Zhou, P., Sridharan, K. and Sapatnekar, S.S.: Congestion-Aware Power Grid Optimization for 3D Circuits Using MIM and CMOS Decoupling Capacitors, *Proc. 2009 Conference on Asia and South Pacific Design Automation*, pp.179–184 (2009).

(Received August 8, 2009)
(Released February 15, 2010)

(Invited by Editor-in-Chief: *Hidetoshi Onodera*)

Jason Cong received his B.S. degree in computer science from Peking University in 1985, his M.S. and Ph.D. degrees in computer science from the University of Illinois at Urbana-Champaign in 1987 and 1990, respectively. Currently, he is a Chancellor's Professor at the Computer Science Department of University of California, Los Angeles, director of Center for Domain-Specific Computing (CDSC), co-director of UCLA/Peking University Joint Research Institute in Science and Engineering, and co-director of the VLSI CAD Laboratory. He also served as the department chair from 2005 to 2008. Dr. Cong's research interests include synthesis of VLSI circuits and systems, programmable systems, novel computer architectures, nano-systems, and highly scalable algorithms. He has over 300 publications in these areas, include four best paper awards. He was elected to an IEEE Fellow in 2000 and ACM Fellow in 2008. Dr. Cong has graduated 25 Ph.D. students. A number of them are now faculty members in major research universities, including Georgia Tech., Purdue, SUNY Binghamton, UCLA, UIUC, and UT Austin. Others are taking key R&D or management positions in major EDA/computer/semiconductor companies, or being founding members of high-tech startups.

Guojie Luo received his B.S. degree from Peking University in 2005, and M.S. degree from University of California, Los Angeles (UCLA), in 2008. He is currently pursuing his Ph.D. degree in computer science at UCLA. He has been working in IBM T.J. Watson Research Center as an intern since 2008. His current research interests include physical design algorithms, 3D integration technology and nonlinear optimization. He is a student member of IEEE and ACM.