Mixed-Size 3D Analytical Placement with Heterogeneous Technology Nodes

Yan-Jen Chen[1], Cheng-Hsiu Hsieh[2], Po-Han Su[2], Shao-Hsiang Chen[2], and Yao-Wen Chang[1,2]
[1]Graduate Institute of Electronics Engineering, National Taiwan University, Taipei 106319, Taiwan
[2]Department of Electrical Engineering, National Taiwan University, Taipei 106319, Taiwan
yjchen@eda.ee.ntu.edu.tw; b09901066@ntu.edu.tw; b08901087@ntu.edu.tw; b09901067@ntu.edu.tw; ywchang@ntu.edu.tw

ABSTRACT

This paper proposes a mixed-size 3D analytical placement framework for face-to-face stacked integrated circuits fabricated with heterogeneous technology nodes and connected by hybrid bonding technology. The proposed framework efficiently partitions a given netlist into two dies and optimizes the positions of each macro, standard cell, and hybrid bonding terminal (HBT). A multi-technology objective function and a multi-technology density penalty calculation process are adopted to handle the heterogeneous-technology-node constraints during mixed-size 3D global placement. Furthermore, a 3D objective function is used to refine the placement result during HBT-cell co-optimization. Our placer achieves the best results for all contest test cases compared with the participating teams at the 2023 CAD Contest at ICCAD on 3D Placement with Macros.

CCS CONCEPTS

• **Hardware** → **Placement**; **3D integrated circuits**.

KEYWORDS

Face-to-face stacked integrated circuits, Heterogeneous integration, Analytical placement

1 INTRODUCTION

Three-dimensional or 2.5D integrated circuits (3D/2.5D ICs) have been recognized as a promising solution to sustain the power/performance/area (PPA) needs of modern technology drivers in the semiconductor industry, such as artificial intelligence, high-performance computing, and automobile electronics. While scaling transistors in two dimensions is reaching physical limits, vertically stacking multiple dies enables 3D/2.5D ICs to achieve higher transistor density and shorter wirelength. As a result, 3D/2.5D ICs can outperform traditional planar ICs in the PPA metric. In addition, 3D/2.5D ICs can facilitate heterogeneous integration, where each die can be fabricated with a different technology node. By appropriately utilizing old and new technology nodes, heterogeneously integrated 3D/2.5D ICs can be manufactured with lower costs while maintaining desired performance. Recent commercial products have demonstrated the benefits of 3D/2.5D IC technology, such as the AMD Ryzen 7-5800X3D [1], the Intel Lakefield Core i5-L16G7 [2], and the TSMC CoWoS [3].

Three major types of interconnection are adopted to connect the stacked dies in 3D/2.5D ICs [4]: (1) through-silicon via (TSV), (2) monolithic inter-tier via (MIV), and (3) hybrid bonding terminal (HBT). TSVs usually come with large micro-scale pitches and parasitics, incurring power and area overheads that may degrade the performance of 3D/2.5D ICs. On the other hand, MIVs utilize nano-scale pitches, which significantly reduce the area cost compared with TSVs. However, vertical white spaces are still needed for MIVs, incurring inevitable area overhead. Compared with TSVs and MIVs, HBTs do not need to reserve spaces on substrates because HBTs connect two dies face-to-face (F2F), resulting in more space for placement and routing. Furthermore, the minimum pitch of HBTs is similar to that of MIVs. Therefore, utilizing HBTs allows a significantly higher 3D/2.5D integration density and is more cost-effective in manufacturing.

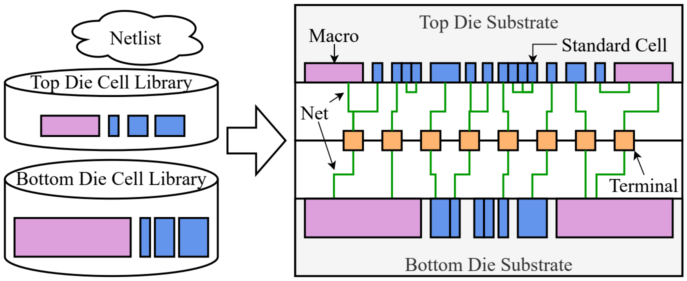


Figure 1: Mixed-size 3D placement problem for an F2F stacked 3D/2.5D IC considering heterogeneous technology nodes. Both macros and standard cells have different shapes in the top die and the bottom die.

The advent of 3D/2.5D ICs introduces new challenges to electronic design automation (EDA) tools, particularly in physical design. Mixed-size placement, which optimizes the positions of circuit *blocks* (*macros and/or standard cells*), is a critical step in the physical design flow. While traditional planar mixed-size placement problems are already complicated due to the size disparity between macros and standard cells, the design complexity explodes in mixed-size 3D placement problems. Additional decisions must be made in mixed-size 3D placement problems, including die partitioning and interconnection placement. Moreover, the constraints of utilizing heterogeneous technology nodes and the effect of the interconnection type must be carefully considered. For example, implementing blocks in dies of different technology nodes results in different dimensions/shapes. As a result, the placement of blocks becomes a chicken-and-egg problem: Which should be determined first, a block's dimension/shape or its placed die? Therefore, mixed-size 3D placement is widely regarded as a significant challenge in producing high-quality 3D/2.5D ICs, demanding advanced methodologies to ensure desired results. Figure 1 shows a mixed-size 3D placement problem for a F2F stacked 3D/2.5D IC considering heterogeneous technology nodes.

1.1 Previous Work

Recent studies on 3D IC placement can be primarily classified into two categories: pseudo-3D and true-3D approaches. Pseudo-3D approaches emulated 3D placement using conventional 2D placers by performing die partitioning on projected 2D placements instead of performing 3D calculations. For instance, Compact2D [5] performed a partitioning-last design flow, which conducted die partitioning from 2D placement results. Snap3D [6] transformed 3D designs into a 2D placement structure by dividing the rows into even and odd ones and carefully setting the placement constraints to place cells in different dies simultaneously. However, the bin-based min-cut partitioning method adopted in these works cannot handle the F2F bonded stacking structure well. Lu *et al.* [7] proposed a graph-neural-network-based die partitioning method to optimize the state-of-the-art monolithic 3D design flow and showed the criticality of the partitioning results. Meanwhile, iPL-3D [8] proposed a bilevel programming model to iteratively update the partitioning result based on a 2D placement prototype.

In contrast, true-3D approaches aimed to optimize block (macro and/or standard-cell) placement in a 3D space, guided by an objective function considering estimated wirelength and interconnection costs. Academic placer NTUplace3-3D [9] and ePlace-3D [10] used analytical methods to perform mixed-size placement for homogeneous TSV-based 3D/2.5D ICs. Chen *et al.* [11] and Liao *et al.* [12] focused on standard-cells-only 3D analytical placement for heterogeneously integrated 3D/2.5D ICs, with the former proposing a logistic-functions-integrated wirelength model

This work was partially supported by AnaGlobe, ASUS, Delta Electronics, Google, Maxeda Technology, TSMC, NSTC of Taiwan under Grant NSTC 110-2221-E-002-177-MY3, NSTC 112-2218-E-002-033, and NSTC 112-2223-E-002-020.

DAC '24, June 23–27, 2024, San Francisco, CA, USA

ACM ISBN 979-8-4007-0601-1/24/06…$15.00
https://doi.org/10.1145/3649329.3657370

and the latter proposing a bistratal wirelength model. Liao *et al.*'s latest true-3D placer outperformed the pseudo-3D placer iPL-3D in tackling heterogeneous 3D placement problems, highlighting the effectiveness of 3D analytical placement.

Despite advancements, previous works faced significant challenges in addressing mixed-size heterogeneous 3D placement for F2F stacked 3D/2.5D ICs. A notable limitation of pseudo-3D placers was their sequential full-chip design approach, constructing the 3D/2.5D chip die-by-die and not fully leveraging 3D/2.5D technology benefits, leading to suboptimal results. True-3D placers, such as NTUPlace3-3D and ePlace-3D, struggled with heterogeneous integration due to their inability to model variations in block shapes across different dies. Furthermore, they aimed to minimize TSV usage due to high manufacturing costs, overlooking the negligible cost of HBT. Chen *et al.*'s and Liao *et al.*'s placers, designed for standard-cells-only placement, did not account for macros' influence. Given these challenges and the suboptimal outcomes of existing placers, there is an urgent need to develop a true-3D mixed-size placer specifically for heterogeneously integrated F2F stacked 3D/2.5D ICs.

1.2 Our Contributions

To remedy the insufficiencies of existing placers, we develop a new mixed-size analytical placement framework for heterogeneously integrated 3D/2.5D ICs. The main contributions of this paper are summarized as follows.

- We propose a mixed-size placement framework for F2F stacked 3D/2.5D ICs fabricated with heterogeneous technology nodes and connected by hybrid bonding technology, which simultaneously solves the netlist partitioning problem and the 3D/2.5D placement problem.
- We propose a multi-technology objective function considering the heterogeneous-technology-node constraints for mixed-size 3D global placement, which effectively guides our placer to optimize the total wirelength and the total HBT cost.
- We propose a multi-technology density penalty calculation procedure. This procedure utilizes logistic functions to model the drastic shape variation caused by changing die assignments. Our calculation procedure provides a precise density estimation during the 3D optimization process.
- We introduce a 3D objective function to further refine the placement result during HBT-cell co-optimization. This function optimizes the exact 3D wirelength and places the HBTs in their optimal region while considering the minimum distance requirement between HBTs.
- Experimental results show that our placer can achieve the best results for all contest test cases in the 2023 CAD Contest at ICCAD on 3D Placement with Macros [13]. Our placer achieves 1.5%, 6.4%, and 9.02% improvements in total scores compared to the top three teams in the contest.

The remainder of this paper is organized as follows. Section 2 introduces the problem formulation. Section 3 describes our placement framework and details our mixed-size 3D placement algorithms. Section 4 reports the experimental results. Finally, Section 5 concludes this paper.

2 PROBLEM FORMULATION

This paper focuses on the mixed-size heterogeneous 3D placement problem, which combines a netlist partitioning problem and a hypergraph $G(V, E)$ placement problem. Given a design netlist, a cell library of the bottom die, and a cell library of the top die, this problem aims to partition the netlist into two dies, insert HBTs to connect split nets, and optimize the positions of each macro, standard cell, and HBT to minimize the total wirelength. Let a set of vertices $V = V_m \cup V_c \cup V_{term}$ represent movable blocks, which can be categorized into macros V_m, standard cells V_c, and HBTs V_{term}. Let a set of hyperedges E represent nets. Our objective is to minimize the scoring function specified in the 2023 CAD Contest at ICCAD on 3D Placement with Macros [13]:

$$W(V_{btm} \cup V_{term}) + W(V_{top} \cup V_{term}) + c_{term}|V_{term}|, \tag{1}$$

where V_{btm}, V_{top}, $W(V_{btm} \cup V_{term})$, $W(V_{top} \cup V_{term})$, and c_{term} denote the bottom-die blocks, top-die blocks, bottom-die total half-perimeter wirelength (HPWL), top-die total HPWL, and cost per HBT, respectively. The constraints of this problem are listed as follows.

- **HBT constraints.** For each net split after partitioning, an HBT must be inserted to connect the signals on the top and bottom die. All HBTs have the same shape and require a minimum spacing between them.
- **Maximum utilization constraints.** The maximum utilization rate of each die is given separately, i.e., the upper bound of the used area in each die is limited.
- **Nonoverlapping constraints.** All blocks cannot overlap. All standard cells must be placed on rows.
- **Technology-node constraints.** The top and bottom dies may be fabricated with different technologies, i.e., each block's width, height, and pin offset may vary in a different die.

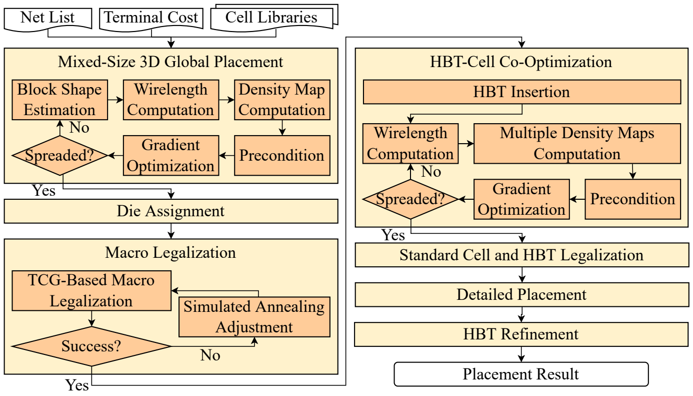


Figure 2: Our mixed-size 3D placement framework.

3 PLACEMENT FRAMEWORK

We develop a seven-stage framework to solve the mixed-size heterogeneous 3D placement problem, as shown in Figure 2. These stages consist of (1) Mixed-Size 3D Global Placement, (2) Die Assignment, (3) Macro Legalization, (4) HBT-Cell Co-Optimization, (5) Standard Cell and HBT Legalization, (6) Detailed Placement, and (7) HBT Refinement. The following sections detail these stages.

3.1 Mixed-Size 3D Global Placement

The first stage optimizes the positions of all blocks in a 3D space. Notice that HBTs are not considered in this stage because the netlist partitioning has not been decided yet. The cell libraries do not contain z-dimensional information such as block depth, which leads to an ill-defined 3D optimization problem. Therefore, we give a reasonable assumption to facilitate the 3D optimization problem.

Assumption 1. *The 3D placement region is* $[0, R_x] \times [0, R_y] \times [0, R_z]$, *where* R_x, R_y, *and* R_z *are the die width, die height, and a user-specified die depth. All macros and standard cells share the same block depth* $\frac{R_z}{2}$.

Under Assumption 1, each movable block is a cube, and each movable block should not be placed outside the placement region. The bottom die can be seen as the lower-half placement region $[0, R_x] \times [0, R_y] \times [0, \frac{R_z}{2})$. Similarly, the top die can be seen as the top-half placement region $[0, R_x] \times [0, R_y] \times [\frac{R_z}{2}, R_z]$. Since the ultimate goal at this stage is placing all blocks in the top or bottom die, the depth of all blocks needs to be set to half the depth of the placement region $\frac{R_z}{2}$ to prevent them from getting stuck in the middle of two dies.

Following the concept of analytical placement, we solve a sequence of unconstrained optimization problems, guided by a multi-technology objective function to determine the position of all movable blocks. The multi-technology objective function is shown in Equation 2:

$$\min \quad W(V_m \cup V_c) + Z(V_m \cup V_c) + \lambda N(V_m \cup V_c), \tag{2}$$

where $W(V_m \cup V_c)$, $Z(V_m \cup V_c)$, $N(V_m \cup V_c)$, and λ are the total wirelength, the weighted HBT cost, the multi-technology density penalty, and the Lagrange multiplier, respectively. The total wirelength approximates the total HPWL while considering the technology-node constraints. Meanwhile, the weighted HBT cost estimates the total number of HBTs and the extra wirelength introduced by HBTs. These two terms approximate Equation 1. Gradient descent is used for optimization. The Lagrange multiplier is increased after each gradient-descent step to spread the movable blocks. Furthermore, a mixed-size preconditioner is used to stabilize the

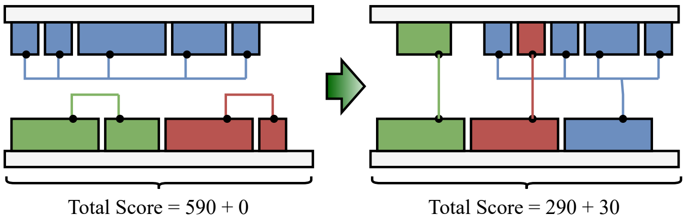


Figure 3: Utilizing three HBTs can result in a much smaller score when the cost per HBT is given as 10.

numerical optimization. The following subsections detail each term in Equation 2 and introduce the mixed-size preconditioner.

3.1.1 *Multi-Technology Wirelength Function.*

To calculate the total wirelength while considering the technology-node constraints, we implemented the multi-technology weighted-average (MTWA) function proposed by Chen [11], which utilizes logistic functions to model the pin-offset variations. The x-component of the MTWA function is shown as follows:

$$\begin{aligned}\hat{p}_i^e(z_i) &= p_{i,1}^e + \frac{p_{i,2}^e - p_{i,1}^e}{1+\exp\left(\frac{-k}{r_2-r_1}\left(z_i - \frac{r_2+r_1}{2}\right)\right)},\\ w_x(e;\mathbf{x},\mathbf{z}) &= \frac{\sum_{v_i\in e}(x_i+\hat{p}_i^e(z_i))\exp(\frac{x_i+\hat{p}_i^e(z_i)}{\gamma})}{\sum_{v_i\in e}\exp(\frac{x_i+\hat{p}_i^e(z_i)}{\gamma})}\\ &\quad - \frac{\sum_{v_i\in e}(x_i+\hat{p}_i^e(z_i))\exp(\frac{-(x_i+\hat{p}_i^e(z_i))}{\gamma})}{\sum_{v_i\in e}\exp(\frac{-(x_i+\hat{p}_i^e(z_i))}{\gamma})},\end{aligned} \tag{3}$$

where e, x_i, z_i, $p_{i,1}^e$, $p_{i,2}^e$, r_1, r_2, k, and γ are a net in the netlist, the x-coordinate, the z-coordinate, the pin x-offset at the bottom die, the pin x-offset at the top die, bottom die's z-coordinate ($\frac{R_z}{4}$), top die's z-coordinate ($\frac{3R_z}{4}$), the user-defined slope constant, and the user-defined smoothing parameter, respectively. The y-component of the MTWA function is defined similarly.

3.1.2 *Weighted HBT Cost.*

Another key aspect of the mixed-size heterogeneous 3D placement problem is that utilizing a minimum cut is no longer the best partitioning strategy. As indicated by Equation 1, there is a trade-off between the total number of HBTs and the total wirelength. Figure 3 illustrates that, given the relatively low cost per HBT, increasing the number of HBTs can be beneficial in reducing the overall wirelength. Besides the constant cost per HBT, inserting HBTs will introduce additional wirelength, which must also be considered.

To accurately estimate the total impact of HBTs, we utilize Equation 4:

$$Z(V_m \cup V_c) = \sum_{e\in E}\left(\frac{c_{term}}{d} + c_e\right)\left(\frac{\sum_{v_i\in e} z_i \exp(\frac{z_i}{\gamma})}{\sum_{v_i\in e}\exp(\frac{z_i}{\gamma})} - \frac{\sum_{v_i\in e} z_i \exp(\frac{-z_i}{\gamma})}{\sum_{v_i\in e}\exp(\frac{-z_i}{\gamma})}\right), \tag{4}$$

where d is a user-specified z-distance between the bottom die and the top die, and c_e is a user-specified weight representing the additional wirelength when inserting HBT into net e. d is set to $\frac{R_z}{2}$ to satisfy Assumption 1. Computing c_e for every net is both time-consuming and complex. We adopt a heuristic that assigns c_e based on the net degree to address this issue. We intend to connect the top and bottom dies with low-degree nets. As a result, 2-pin nets will have a lower weight than other nets.

3.1.3 *Multi-Technology Density Penalty Calculation.*

In updating the density penalty, it is vital to incorporate the technology-node constraints and the maximum utilization constraints. To address this, we develop a new 3D density update procedure inspired by the eDensity from the ePlace series [10][14]. eDensity utilizes an electrostatic system to represent the nonoverlapping constraints, treating each movable block $v_i \in V_m \cup V_c$ as a positive charge q_i. In this analogy, the density penalty corresponds to the electrostatic system's potential energy $N(V_m \cup V_c) = \sum q_i\phi_i$, while the gradient of the density penalty is modeled as an electric force. The numerical solutions for potential energy and the electric field are derived by solving Poisson's Equation using spectral methods and the fast Fourier transform (FFT). Let the 3D placement region be uniformly divided by a $M_x \times M_y \times M_z$ grid. Let the frequency indexes be denoted as $(\omega_j, \omega_k, \omega_l) = (\frac{\pi j}{R_x}, \frac{\pi k}{R_y}, \frac{\pi l}{R_z})$. The numerical solution of the electrostatic system is shown as follows:

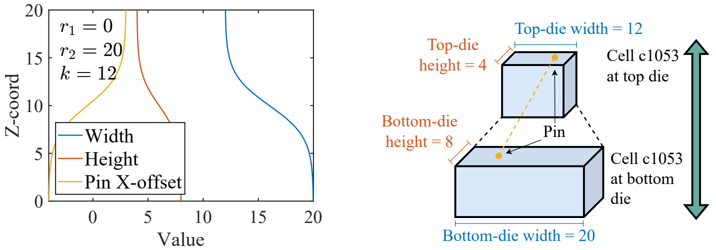


Figure 4: The pin-offset, width, and height of standard cells are updated using corresponding logistic functions. Macros are treated with the same method.

$$a_{j,k,l} = \frac{1}{M_x M_y M_z}\sum_{x,y,z}\rho(x,y,z)\cos(\omega_j x)\cos(\omega_k y)\cos(\omega_l z), \tag{5}$$

$$\phi(x,y,z) = \sum_{j,k,l}\frac{a_{j,k,l}}{\omega_j^2+\omega_k^2+\omega_l^2}\cos(\omega_j x)\cos(\omega_k y)\cos(\omega_l z), \tag{6}$$

$$\begin{aligned}\xi_x(x,y,z) &= \sum_{j,k,l}\frac{a_{j,k,l}\omega_j}{\omega_j^2+\omega_k^2+\omega_l^2}\sin(\omega_j x)\cos(\omega_k y)\cos(\omega_l z),\\ \xi_y(x,y,z) &= \sum_{j,k,l}\frac{a_{j,k,l}\omega_k}{\omega_j^2+\omega_k^2+\omega_l^2}\cos(\omega_j x)\sin(\omega_k y)\cos(\omega_l z),\\ \xi_z(x,y,z) &= \sum_{j,k,l}\frac{a_{j,k,l}\omega_l}{\omega_j^2+\omega_k^2+\omega_l^2}\cos(\omega_j x)\cos(\omega_k y)\sin(\omega_l z),\end{aligned} \tag{7}$$

where $a_{j,k,l}$, $\rho(x,y,z)$, $\phi(x,y,z)$, and $\xi(x,y,z)$ are the density coefficient, density function, potential function, and electric field, respectively.

Unlike ePlace-3D, we update all movable block's width and height before every time calculating Equation 5. We model height variation using logistic functions, shown as follows:

$$\hat{h}_i(z_i) = h_{i,1} + \frac{h_{i,2} - h_{i,1}}{1+\exp\left(\frac{-k}{r_2-r_1}\left(z_i - \frac{r_2+r_1}{2}\right)\right)}, \tag{8}$$

where $h_{i,1}$ ($h_{i,2}$) is the bottom-die (top-die) block height. Width variation is modeled similarly. Figure 4 showcases our block shape variation model. As a result, we can accurately estimate the 3D density distribution while considering the technology-node constraints.

Meanwhile, we treat the maximum utilization constraints as soft constraints at this stage. Unlike ePlace-3D, which allows all fillers to move freely within the 3D placement region without considering maximum utilization constraints, our method inserts two types of fillers into the placement region. These fillers emulate the maximum utilization constraints of the bottom (first-type fillers) and top (second-type fillers) dies. We calculate the total area of these fillers as follows:

$$\begin{aligned}A_1 &= R_x R_y (1 - u_{btm}),\\ A_2 &= R_x R_y (1 - u_{top}),\end{aligned} \tag{9}$$

where A_1, A_2, u_{btm}, and u_{top} denote the total areas of the first and second-type fillers and the maximum utilization rates of the bottom die and top die, respectively. Under Assumption 1, the depth of all fillers is set to $\frac{R_z}{2}$. All fillers are initially placed within their respective dies. During the optimization, the filler's z-gradient is set to zero to prevent moving to other dies. As a result, fillers will act as pre-occupied space. Once a die's maximum utilization rate is exceeded, the fillers will push movable blocks toward another die.

3.1.4 *Mixed-Size Preconditioner.*

To effectively optimize Equation 2 for large-scale circuits, we implement a mixed-size preconditioner designed to stabilize the optimization process. Figure 5 shows a plateau stage in the optimization process, which brings overhead and potentially degrades the solution quality. The optimization process is monitored by the overflow

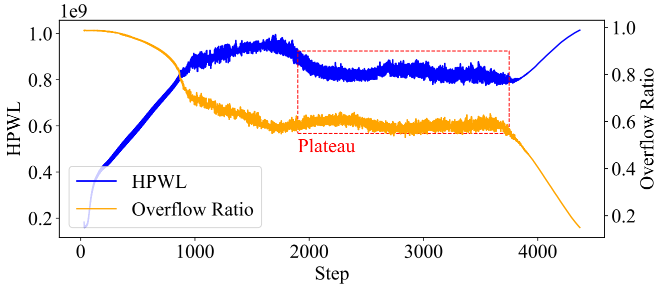


Figure 5: A plateau is observed while placing case4 in the 2023 CAD Contest at ICCAD on 3D Placement with Macros [13].

ratio, the ratio of the overlapped volume to the design's total movable volume. In the early phase of the optimization process, the gradients of macros are significantly larger than those of standard cells, primarily because macros are connected to a larger number of nets. This gap in gradient magnitude causes considerable movement in the macros, potentially disrupting the intended optimization direction.

We adopt the following mixed-size preconditioner:

$$P_i(v_i) = \begin{cases} \max(1, \#\text{pins}(v_i) + \lambda\ \text{vol}(v_i))^{-1}, & \text{if } v_i \text{ is a macro,} \\ \max(1, \lambda\ \text{vol}(v_i))^{-1}, & \text{otherwise,} \end{cases} \qquad (10)$$
$$\nabla f_{pre} = \nabla f \odot P,$$

which uses the number of pins to estimate the second-order derivative of the wirelength function, while the block volume is used to estimate the second-order derivative of the density penalty. The difference between Equation 10 and the previous work [10] is that we only estimate the second-order derivative of the wirelength function for macros. In practice, our mixed-size preconditioner reduces the magnitude of the macros' gradient and stabilizes the early optimization phase.

3.2 Die Assignment

The die assignment stage partitions the netlist into two dies according to the 3D placement obtained from the previous stage while considering the maximum utilization constraints. The specific subproblem we address is minimizing the z-direction displacement while the maximum utilization constraints are satisfied, as shown in Equation 11:

$$\begin{aligned} \min_{\hat{z}_i} \quad & \sum_{v_i \in V_m \cup V_c} (1 - \hat{z}_i) z_i + \hat{z}_i (R_z - z_i), \\ \text{s.t.} \quad & \sum_{v_i \in V_m \cup V_c} \hat{z}_i \ \text{area}_{top}(v_i) \le u_{top}, \\ & \sum_{v_i \in V_m \cup V_c} (1 - \hat{z}_i) \ \text{area}_{btm}(v_i) \le u_{btm}, \\ & \hat{z}_i \in \{0, 1\}, \end{aligned} \qquad (11)$$

where z_i is the z-coordinate determined in the previous stage, and $\hat{z}_i$ stands for the decision variable for die assignment, with 0 indicating the bottom die and 1 indicating the top die.

To solve this subproblem, we propose a greedy algorithm, detailed in Algorithm 1. This algorithm comprises two phases: a macro partitioning stage and a standard-cell partitioning stage. We prioritize macros due to their greater influence on the final solution than standard cells. Since the 3D placement prototype is almost partitioned, blocks with higher z-coordinates are likelier to be assigned to the top die. Consequently, we sort the blocks in non-increasing order of their z-coordinates and then assign them to the closest die. When the maximum utilization rate of a die is exceeded, subsequent blocks are allocated to the alternate die. Experimental result shows that we can always find a feasible solution, demonstrating the high quality of the 3D placement and the effectiveness of our algorithm in the previous stage.

Algorithm 1 Partitioning according to 3D placement.

```
Input: Macros $V_m$, standard cells $V_c$, z-coordinates $z_i$, and bottom (top)
    die maximum utilization rate $u_{btm}$ ($u_{top}$)
Output: Bottom-die blocks $V_{btm}$ and top-die blocks $V_{top}$
  $V_{btm} \leftarrow \{\}, V_{top} \leftarrow \{\}$;
  Partition($V_m$);
  Partition($V_c$);
  raise an error if $u_{btm}$ or $u_{top}$ is violated;
  Function Partition($V$):
    Sort($V$);
    foreach $v_i \in V$ do
      if place $v_i$ at the top die will violate $u_{top}$ then
        $V_{btm} \leftarrow V_{btm} \cup v_i$;
      else if place $v_i$ at the bottom die will violate $u_{btm}$ then
        $V_{top} \leftarrow V_{top} \cup v_i$;
      else if $z_i \le R_z - z_i$ then
        $V_{btm} \leftarrow V_{btm} \cup v_i$;
      else
        $V_{top} \leftarrow V_{top} \cup v_i$;
```

3.3 Macro Legalization

In this stage, macro legalization is performed die-by-die to eliminate overlaps between macros. We utilize the transitive-closure-graph-based (TCG-based) macro legalization algorithm [15], and apply it to each die sequentially based on the results from Section 3.2. In cases where the macros result in an infeasible TCG, a simulated-annealing-based algorithm [14] is employed to modify the positions of the macros. After completing the macro legalization process, all macros are fixed in positions.

3.4 HBT-Cell Co-Optimization

An accurate 3D analytical placement is used to refine the solution from Section 3.3 because die assignment and macro legalization disturb the 3D placement prototype obtained from Section 3.1. Firstly, HBTs are inserted according to the result from Section 3.3. After HBT insertion, a 3D objective function is applied to co-optimize the positions of HBTs and standard cells, shown as follows:

$$\min \quad W(V_{btm}, V_{top}, V_{term}) + \lambda_s{}^{\top} \mathbf{N}_s(V_{btm}, V_{top}, V_{term}), \qquad (12)$$

where $W(V_{btm}, V_{top}, V_{term})$, λ_s, and $\mathbf{N}_s(V_{btm}, V_{top}, V_{term})$ are the exact 3D wirelength, the vector of Lagrangian multipliers $\langle \lambda_{btm}, \lambda_{top}, \lambda_{term} \rangle$, and the vector of density penalties $\langle N(V_{btm}), N(V_{top}), N(V_{term}) \rangle$, respectively. Similar to Section 3.1, gradient descent is used for optimization. The following subsections detail our terminal insertion method and each term in Equation 12.

3.4.1 HBT Insertion. For each net connecting two dies, we modify the netlist as follows: the original net e is divided into a bottom-die net e_{btm} and a top-die net e_{top}, with an HBT inserted and connected to e_{btm} and e_{top}. HBTs are initially positioned at the center of their optimal region, as outlined by Liao *et al.* [12]. The optimal region $[x_t^-, x_t^+]$ of a net e on the x-direction is shown as follows:

$$\begin{aligned} x_{btm}^- &= \min_{v_i \in e_{btm}} (x_i), \quad & x_{btm}^+ &= \max_{v_i \in e_{btm}} (x_i), \\ x_{top}^- &= \min_{v_i \in e_{top}} (x_i), \quad & x_{top}^+ &= \max_{v_i \in e_{top}} (x_i), \end{aligned} \qquad (13)$$

$$\begin{aligned} x_t^- &= \min \left\{ \min \left\{ x_{btm}^+, x_{top}^+ \right\}, \max \left\{ x_{btm}^-, x_{top}^- \right\} \right\}, \\ x_t^+ &= \max \left\{ \min \left\{ x_{btm}^+, x_{top}^+ \right\}, \max \left\{ x_{btm}^-, x_{top}^- \right\} \right\}. \end{aligned} \qquad (14)$$

The optimal region on the y-direction is calculated similarly.

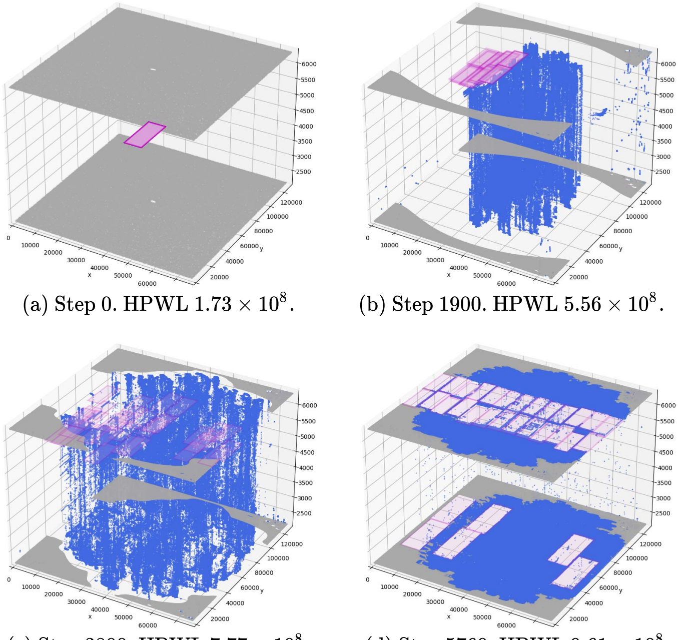
3.4.2 Exact 3D Wirelength Function. The exact wirelength of 3D/2.5D placement is the sum of the bottom-die HPWL and the top-die HPWL, defined in Equation 1 and shown as follows:

$$W(V_{btm}, V_{top}, V_{term}) = W(V_{btm} \cup V_{term}) + W(V_{top} \cup V_{term}), \qquad (15)$$

while HBT positions are considered during the calculation. Similar to the 3D wirelength function, we use the WA model [9] to approximate the non-differentiable HPWL. The x-component of the bottom-die wirelength function is shown as follows.

Table 1: The benchmark statistics of the 2023 CAD Contest at ICCAD on 3D Placement with Macros.

Circuit	#Macros	#Cells	#Nets	u_{btm}	u_{top}	c_{term}	Diff Tech
case1	3	5	6	0.9	0.8	10	Yes
case2	6	13901	19547	0.8	0.8	10	No
case2h1	6	13901	19547	0.8	0.8	10	Yes
case2h2	6	13901	19547	0.8	0.8	10	Yes
case3	34	124231	164429	0.8	0.8	10	Yes
case3h	34	124231	164429	0.8	0.8	10	Yes
case4	32	740211	758860	0.8	0.8	10	Yes
case4h	32	740211	758860	0.8	0.8	10	Yes



(a) Step 0. HPWL 1.73×10^8.

(b) Step 1900. HPWL 5.56×10^8.

(c) Step 3800. HPWL 7.77×10^8.

(d) Step 5760. HPWL 9.61×10^8.

Figure 6: Our mixed-size 3D global placement on case4. Macros, standard cells, and fillers are denoted by purple, blue, and gray, respectively. The block depth has been omitted to improve visual clarity.

$$W_x(V_{btm} \cup V_{term}) = \sum_{e \in E_{btm}} \left(\frac{\sum_{v_i \in e} x_i \exp(\frac{x_i}{\gamma})}{\sum_{v_i \in e} \exp(\frac{x_i}{\gamma})} - \frac{\sum_{v_i \in e} x_i \exp(\frac{-x_i}{\gamma})}{\sum_{v_i \in e} \exp(\frac{-x_i}{\gamma})} \right), \tag{16}$$

where E_{btm} is the set of the bottom-die nets. The y-component of the bottom-die and top-die wirelength function is defined similarly. Notably, the gradient of wirelength relative to the HBTs' coordinates directs the HBTs to move toward their optimal region.

3.4.3 Layer-by-Layer 3D Density Penalties. This stage considers three constraints: the nonoverlapping constraints in the top die, the nonoverlapping constraints in the bottom die, and the minimum spacing constraint between HBTs. These constraints are modeled using the 2D eDensity model [14]. The top-die blocks, bottom-die blocks, and HBTs each contribute to three separate density penalties ($N(V_{btm})$, $N(V_{top})$, and $N(V_{term})$), with their respective Lagrange multipliers updated independently. Unlike the approach of Chen *et al.* [11], which does not consider the density of the HBTs, our formulation includes the HBT-density penalty $N(V_{term})$ to prevent the potential increase in wirelength caused by a congested HBT placement.

Considering the minimum spacing constraint between each HBT, we pad the HBT shape as follows:

$$\bar{w_t} = w_t + d_t, \tag{17}$$

where d_t, w_t, and $\bar{w_t}$ are the minimum spacing requirement, original HBT width, and padded HBT width. The height is padded similarly. Since the HBT density penalty is calculated according to the padded shape, the minimum spacing constraint is naturally considered during the optimization.

3.5 Standard Cell and HBT Legalization

The standard cells and HBTs are legalized die-by-die. We employ row-based algorithms such as Abacus [16] and Tetris [17] for this task, ultimately selecting the outcome that yields the minimum HPWL. Special attention is paid to the unique constraints of HBTs: they are legalized with additional padding space to ensure compliance with the minimum distance requirements, thereby avoiding any constraint violations.

3.6 Detailed Placement

After the previous stage, cell matching [18] and cell swapping [19] further refine the standard cell positions and HBT positions.

3.7 HBT Refinement

Unlike standard cells, HBTs do not need to be placed on rows. When using row-based legalization, unwanted displacement of HBTs can occur. To enhance HBT placement, we first verify if they are within their optimal region. If not, we systematically search adjacent legal points, prioritizing those with lower HPWL to minimize deviation from ideal positions. Once relocation efforts fail, the HBT remains in its original location.

4 EXPERIMENTAL RESULTS

We compared our placement framework with the top 3 teams in the 2023 CAD Contest at ICCAD on 3D Placement with Macros [13]. Table 1 shows the statistics of the contest benchmark suite. The benchmarks include one toy case and seven mixed-size designs. Most of the designs contain heterogeneous technology nodes.

Our framework was implemented in C++ programming language with OpenMP multi-thread library. All experiments were conducted on a Linux workstation with AMD Ryzen 3990X 2.9GHz processors and 128 GB of memory. The placement results were validated and scored by the contest evaluator, where the score was calculated according to Equation 1. For a fair comparison, the maximum number of threads was limited to eight, the same as the contest setup.

Table 2 summarizes the experimental results comparing our placement framework with the top three teams. These results were obtained by executing their binary files on our machine. The data reveals that our placer outperformed the first, second, and third-place teams by 1.5%, 6.4%, and 9.02% on the total score, respectively. In the most challenging case, case 4h, we achieved a 2.96%, 5.3%, and 9.91% improvement in score compared with the top 3 teams. Our placer achieved a smaller (better) score than the first and third-place teams while utilizing more HBTs. This success is attributed to our mixed-size 3D global placement, which utilized the multi-technology objective function to optimize the wirelength and total terminal cost. While the second-place team employed a partitioning-based flow, our placer significantly outperformed their results in terms of score, thereby highlighting the superiority of our true-3D placement approach. Our placer consistently produced the best results across all contest test cases, encompassing various problem sizes, in reasonable runtime. Notably, the second-place team utilized the partitioning-based flow, lacking 3D computation, which naturally resulted in the fastest runtime in all nontrivial cases.

Figure 6 shows the snapshots of our mixed-size 3D global placement framework performed on case 4. All blocks were centered in the beginning according to the result of initial placement. In the early phase, blocks mainly spread on the z-axis, which implicitly conducted a preliminary die assignment. In the next phase, blocks spread toward the x and y directions, while the exchange of z positions between blocks still frequently occurred. In the end, all macros and standard cells were stabled in a preferred die. Blocks were nearly separated to discrete along the z-axis, with a few standard cells still switching their layers. All blocks were co-optimized during the process, preventing them from getting stuck in the local optimal and giving us more chances to reach the desired solutions.

We conducted an ablation study to assess the impact of the HBT-cell co-optimization stage in our placement process, as summarized in Table 3. The data shows that the HBT-cell co-optimization contributed to a 3.85% improvement. These findings revealed that the overhead resulting from the die partitioning and macro legalization stages could not be mitigated without this co-optimization stage, justifying the effectiveness and necessity of our algorithm and the 3D objective.

Table 2: Comparison among the top 3 teams and our placer in the contest. The score was evaluated according to Equation 1.

Circuit	1st place			2nd place			3rd place			Our Placer		
	Score	#HBTs	Time(s)	Score	#HBTs	Time(s)	Score	#HBTs	Time(s)	Score	#HBTs	Time(s)
case1	**113**	1	1	214	5	59	207	3	14	**113**	1	1
case2	16506066	1523	101	16337444	993	27	16843851	2113	116	**16026869**	1589	177
case2h1	18123044	120	52	19076604	821	33	21400624	2386	130	**17695111**	176	74
case2h2	18124483	120	54	19225916	821	34	21208927	2359	129	**17701682**	176	77
case3	98928220	22189	547	105539748	26397	85	110528173	21795	548	**98737450**	22189	519
case3h	122459408	18761	235	123037575	5027	83	116194632	23279	564	**115694517**	25444	196
case4	1047852542	160993	3063	1119507941	187417	486	1133976910	106904	4381	**1045958081**	123345	3149
case4h	656528147	156586	2321	672803779	168600	484	707172707	144292	3656	**637119983**	173224	2266
Sum	1978522023	360293	6374	2075529221	390081	1291	2127326031	303131	9538	**1948933806**	346144	6459
Comp.	1.0000	1.0000	1.0000	1.0490	1.0827	0.2025	1.0752	0.8413	1.4964	**0.9850**	0.9607	1.0133

Table 3: The result of the ablation studies on the contest benchmarks. The HBT-cell co-optimization stage was removed to test the effectiveness of this stage.

Circuit	Ours w/o co-opt.			Ours w/ co-opt.		
	Score	#HBTs	Time(s)	Score	#HBTs	Time(s)
case1	**113**	1	1	**113**	1	1
case2	17775403	1589	173	**16026869**	1589	177
case2h1	20357045	176	46	**17695111**	176	74
case2h2	20367511	176	48	**17701682**	176	77
case3	112059825	22189	451	**98737450**	22189	519
case3h	124620362	25444	170	**115694517**	25444	196
case4	1077385032	123345	2598	**1045958081**	123345	3149
case4h	651379395	173224	1829	**637119983**	173224	2266
Sum	2023944686	346144	5316	**1948933806**	346144	6459
Comp.	1.0385	1.0000	0.8230	**1.0000**	1.0000	1.0000

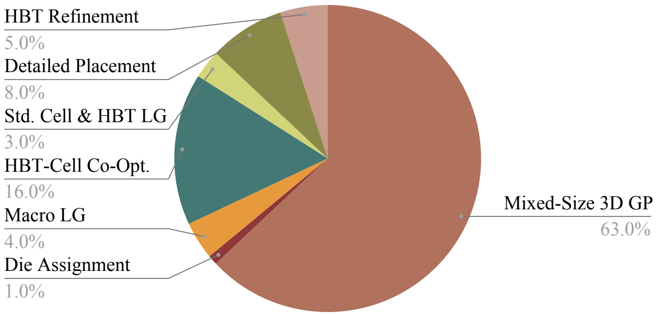


Figure 7: The runtime breakdown on case4h. GP is the abbreviation for Global Placement, and LG stands for Legalization.

Figure 7 shows the runtime breakdown on the benchmark for our placer. Global placement takes 63% of the time, which is the main step of our framework. The next time-consuming step is HBT-cell co-optimization (16%), and the third is detailed placement (8%). Macro legalization, standard cell and HBT legalization, and HBT refinement take no more than 5% of the time each.

5 CONCLUSIONS

We have proposed a high-quality mixed-size 3D placement framework to solve the heterogeneous 3D placement problem. Our framework consists of seven stages, among which the following two stages are most critical: (1) The mixed-size 3D global placement stage, which has employed the multi-technology objective function and the multi-technology density penalty calculation process to accurately estimate wirelength and density distribution while considering the heterogeneous-technology-node constraints and maximum utilization constraints, and (2) The HBT-cell co-optimization stage, which has refined the placement result by simultaneously adjusting the positions of standard cells and HBTs, using a 3D objective for guidance. Our experimental results have demonstrated that our framework is superior to the participating teams in the 2023 CAD Contest at ICCAD on 3D Placement with Macros. Specifically, our placer has shown improvements of 1.5%, 6.4%, and 9.02% in total scores over the first, second, and third-place teams, respectively. Moreover, we have achieved score enhancements of up to 2.96%, 5.3%, and 9.91% on the most challenging test case compared to the top three teams. These results have justified our framework's great ability to handle technology-node constraints in 3D IC placement.

REFERENCES

[1] J. Wuu, R. Agarwal, M. Ciraula, C. Dietz, B. Johnson, D. Johnson, R. Schreiber, R. Swaminathan, W. Walker, and S. Naffziger, "3D V-cache: the implementation of a hybrid-bonded 64MB stacked cache for a 7nm x86-64 CPU," in *Proceedings of IEEE International Solid-State Circuits Conference*, Virtual Event, February 2022.

[2] W. Gomes, S. Khushu, D. B. Ingerly, P. N. Stover, N. I. Chowdhury, F. O'Mahony, A. Balankutty, N. Dolev, M. G. Dixon, L. Jiang *et al.*, "8.1 Lakefield and mobility compute: A 3D stacked 10nm and 22FFL hybrid processor system in 12× 12mm 2, 1mm package-on-package," in *Proceedings of IEEE International Solid-State Circuits Conference*, San Francisco, California, February 2020.

[3] TSMC, "The chronicle of CoWoS." [Online]. Available: https://3dfabric.tsmc.com/english/dedicatedFoundry/technology/cowos.htm

[4] J. Kim, L. Zhu, H. M. Torun, M. Swaminathan, and S. K. Lim, "Micro-bumping, hybrid bonding, or monolithic? a PPA study for heterogeneous 3D IC options," in *Proceedings of ACM/IEEE Design Automation Conference*, San Francisco, California, December 2021.

[5] B. W. Ku, K. Chang, and S. K. Lim, "Compact-2D: A physical design methodology to build two-tier gate-level 3-D ICs," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 39, no. 6, pp. 1151–1164, 2019.

[6] P. Vanna-Iampikul, C. Shao, Y.-C. Lu, S. Pentapati, and S. K. Lim, "Snap-3D: A constrained placement-driven physical design methodology for face-to-face-bonded 3D ICs," in *Proceedings of International Symposium on Physical Design*, Virtual Event, March 2021.

[7] Y.-C. Lu, S. S. Kiran Pentapati, L. Zhu, K. Samadi, and S. K. Lim, "TP-GNN: A graph neural network framework for tier partitioning in monolithic 3D ICs," in *Proceedings of ACM/IEEE Design Automation Conference*, San Francisco, California, July 2020.

[8] X. Zhao, S. Chen, Y. Qiu, J. Li, Z. Huang, B. Xie, X. Li, and Y. Bao, "iPL-3D: A novel bilevel programming model for die-to-die placement," in *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, San Francisco, California, October/November 2023.

[9] M.-K. Hsu, V. Balabanov, and Y.-W. Chang, "TSV-aware analytical placement for 3-D IC designs based on a novel weighted-average wirelength model," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 32, no. 4, pp. 497–509, 2013.

[10] J. Lu, H. Zhuang, I. Kang, P. Chen, and C.-K. Cheng, "ePlace-3D: Electrostatics based placement for 3D-ICs," in *Proceedings of International Symposium on Physical Design*, Santa Rosa, California, April 2016.

[11] Y.-J. Chen, Y.-S. Chen, W.-C. Tseng, C.-Y. Chiang, Y.-H. Lo, and Y.-W. Chang, "Late breaking results: Analytical placement for 3D ICs with multiple manufacturing technologies," in *Proceedings of ACM/IEEE Design Automation Conference*, San Francisco, California, July 2023.

[12] P. Liao, Y. Zhao, D. Guo, Y. Lin, and B. Yu, "Analytical die-to-die 3D placement with bistratal wirelength model and GPU acceleration," *arXiv preprint arXiv:2310.07424*, 2023.

[13] K.-S. Hu, H.-Y. Chi, I.-J. Lin, Y.-H. Wu, W.-H. Chen, and Y.-T. Hsieh, "2023 ICCAD CAD contest problem B: 3D placement with macros," in *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, San Francisco, California, October/November 2023.

[14] J. Lu, H. Zhuang, P. Chen, H. Chang, C.-C. Chang, Y.-C. Wong, L. Sha, D. Huang, Y. Luo, C.-C. Teng *et al.*, "ePlace-MS: Electrostatics-based placement for mixed-size circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 34, no. 5, pp. 685–698, 2015.

[15] H.-C. Chen, Y.-L. Chuang, Y.-W. Chang, and Y.-C. Chang, "Constraint graph-based macro placement for modern mixed-size circuit designs," in *Proceedings of IEEE/ACM International Conference on Computer-Aided Design*, San Jose, California, November 2008.

[16] P. Spindler, U. Schlichtmann, and F. M. Johannes, "Abacus: Fast legalization of standard cell circuits with minimal movement," in *Proceedings of International Symposium on Physical Design*, Portland, Oregon, April 2008.

[17] D. Hill, "Method and system for high speed detailed placement of cells within an integrated circuit design," Apr. 9 2002, U.S. Patent 6 370 673.

[18] T.-C. Chen, Z.-W. Jiang, T.-C. Hsu, H.-C. Chen, and Y.-W. Chang, "NTUplace3: An analytical placer for large-scale mixed-size designs with preplaced blocks and density constraints," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 7, pp. 1228–1240, 2008.

[19] Z.-W. Jiang, T.-C. Cheny, T.-C. Hsuy, H.-C. Chenz, and Y.-W. Chang, "NTUplace2: A hybrid placer using partitioning and analytical techniques," in *Proceedings of International Symposium on Physical Design*, San Jose, California, April 2006.