

Thermal Via Planning for 3-D ICs *

Jason Cong
Computer Science Department, UCLA
Los Angeles, CA 90095
cong@cs.ucla.edu

Yan Zhang
Computer Science Department, UCLA
Los Angeles, CA 90095
zhangyan@cs.ucla.edu

ABSTRACT

Heat dissipation is one of the most serious challenges in 3-D IC designs. One effective way of reducing circuit temperature is to introduce thermal through-the-silicon (TTS) vias. In this paper, we extended the TTS-via planning in a multilevel routing framework as in [7], but use a much enhanced TTS-via planning algorithm. We formulate the TTS-via minimization problem with temperature constraints as a constrained nonlinear programming problem (NLP) based on the thermal resistive model and develop an efficient heuristic algorithm, named m-ADVP, which solves a sequence of simplified via planning subproblems in alternating direction in a multilevel framework. The vertical via distribution is formulated as a convex programming problem, and the horizontal via planning is based on two efficient techniques: *path counting* and *heat propagation*. Experimental results show that the m-ADVP algorithm is more than 200× faster than the direct solution to the NPL formulation for via planning with very similar solution quality (within 1% of TS-vias count). However, compared to a recent work of multilevel TS-via planning algorithm based on temperature profiling [7], our algorithm can reduce the total TS-via number by over 68% for the same required temperature with similar runtime.

Categories and Subject Descriptors

T3.2 [Design Tools Track]: [Global and Detailed Routing]

General Terms

thermal-driven, 3-D IC, routing, via planning

1. INTRODUCTION

3-D IC architecture has recently drawn much attention because of its potential for reducing the interconnect delay and integrating heterogenous components together for system-on-a-chip (SoC) designs [2]. One major challenge of 3-D IC circuit design is thermal problem due to the higher power density and low thermal conductivity of the inter-layer dielectrics (ILD) [13].

There are two kinds of circuit cooling schemes for 3D ICs, including heat sink optimization methods and internal heat distribution optimization methods. Heat sink optimization methods, such as air cooling by electrical fans, micro-channel cooling at heat sinks, etc., are often targeted at cooling down

the heat sink of a chip. In 3D ICs, however, because of the poor thermal conductivity of the ILD layers, the heat generated by the devices cannot be effectively dissipated toward the heat sinks. Studies by Raman and Turowski [16] also show that conventional cooling package techniques, such as cooling fans, might not prove effective for alleviating the thermal problem. Chip-level cooling schemes, on the other hand, often assume a perfect heat sink which can always be cooled down to certain low temperatures, and change the internal structure of the circuit for better heat dissipation, such as the locations of the devices, thermal TS-via insertion, micro-channel insertion [8], etc.

Since macro blocks and cells are the major heat sources in a circuit, the positions of the blocks and cells will affect the circuit temperature. One kind of 3-D IC thermal optimization effort is to add temperature as another optimization objective during 3-D floorplanning [6] or 3-D placement [9]. However, even after thermal-driven floorplanning or thermal-driven placement, the maximum on-chip temperature can still be as high as 150°C [6, 9], which is too high for a circuit to operate properly. Another cooling scheme proposed for 3-D IC is to insert micro-channels into the circuit for better lateral thermal dissipation [8]. However, micro-channel cooling scheme is usually very costly to fabricate, and the micro-channels will also create large obstacles for the TS-vias. The simulation results in [16] also show that metal wires and vias do not have much impact on 3-D circuit temperature either. However, “*through-the-silicon*” vias (TS-vias, shown in Figure 1) are very effective at heat dissipation. When the *signal through-the-silicon* (STS) via number is not enough, usually *thermal through-the-silicon* (TTS) vias will be introduced to the circuit in order to reduce the chip temperature to a satisfactory level. TTS-via insertion cooling schemes have been studied by different groups at both the packaging and the chip level [3, 14].

However, TS-vias are usually etched or drilled through device layers by special techniques [11] and are costly to fabricate. Large numbers of the TS-vias will degrade the yield of the final chip. Also, under current technologies, the common TS-via pitch is very large compared to that of regular metal wires, usually around 5~10μm, which is even larger than a standard cell. In 3-D IC structures, TS-vias are usually placed at the whitespace between the macro blocks or cells, as shown in Figure 1, so the number of TS-vias will not only affect the routing resource but also the overall chip or package areas. Therefore, the number of TTS-vias inserted into the circuit needs be minimized while maintaining the temperature constraint.

The process of minimizing the TS-via number with temperature constraint is called *TS-via planning*. The TS-via plan-

*This research is supported by DARPA under Prime Contract DAAH01-03-C-R193, and CFD Research Corp under Subcontract 03-102.

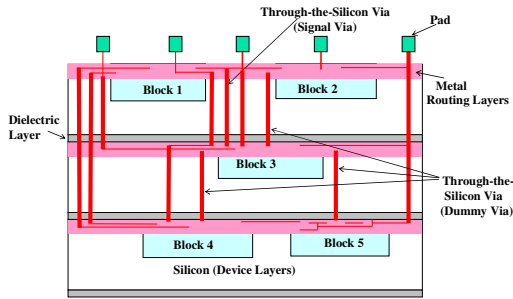


Figure 1: Cross-Section of a 3-D IC Stack

ning problem can be considered at different phases of physical design, such as during or after floorplanning or placement [10], during global routing [7] or after global routing [7]. Before global routing, the number and distribution of the STS-vias are still unknown, so the TTS-via planning at this stage usually pessimistically inserts more TTS-vias than necessary. Also, it will be hard for routers to reserve whitespace for later TTS-via insertion, so it will be very difficult to insert TTS-vias after the routing is completed. Therefore, we choose to plan the TTS-vias *during* the routing process as in [7].

In this paper we extend the TS-via planning in a multilevel routing framework as in [7] by improving the TS-via planning heuristic using a much more effective TS-via minimization technique, named *m-ADVP*. We formulate the TTS-via minimization problem as a constrained NLP and solve it by solving a sequence of simplified TS-via planning subproblems in alternating directions in a multilevel framework. Our contributions include: (1) we propose a fast and effective multilevel alternating direction TTS-via planning algorithm (*m-ADVP*) which iteratively distributes TTS-vias vertically and horizontally; (2) we formulate the vertical TTS-via distribution as a convex programming problem with an analytical optimal solution for cases with no capacity constraint; (3) we also propose two effective techniques, *path counting* and *heat propagation* for the horizontal TTS-via distribution; (4) our algorithm is based on the heat dissipating path analysis instead of explicit thermal profiling, so the modeling and the optimization are combined into one multilevel framework. We also solve the NLP problem directly through a conjugate gradient penalty function-based package.

The remainder of this paper is organized as follows. Section 2 reviews the multilevel routing and TS-via planning framework in [7]. Section 3 introduces the resistive thermal model and the NLP formulation of the TTS-via planning problem. The *m-ADVP* algorithm is presented in Section 4. Experimental results are listed in Section 5, and Section 6 concludes the paper.

2. REVIEW OF THE MULTILEVEL TS-VIA PLANNING FRAMEWORK

Previous work [7] has shown that a multilevel scheme is effective in TS-via optimization, so we also integrate our TS-via optimization scheme into a multilevel routing framework. In this section, we will briefly review the multilevel routing and TS-via planning framework in [7].

The multilevel routing and TS-via planning framework is composed of three stages: *recursive coarsening*, *initial solution generation*, and *level-to-level refinement*, as shown in

Figure 2.

The 3-D circuit stack is first divided into tiles. Routing resources, including the TS-via capacity, and the heat flow of each tile is calculated. Then, during the coarsening process, a series of similar problems with larger tiles and smaller problem size is generated. Each coarsened problem corresponds to a different level. Routing resources and power density values of the coarsened tiles are also computed at each level.

The TTS-via number needed is first estimated at the coarsest level. The TTS-via planning algorithm is first called at the coarsest level after the initial routing tree generation to reduce temperature as much as possible. The refinement process then starts with the coarsest level solution and moves from the coarser level to the finer level in reverse order from the coarsening process. During each refinement, the TS-via planning algorithm will repeatedly refine the TS-via distribution within each coarser-level tile stack so that the coarser level solution is preserved. If T_{input} is not met at the end of refinement, a proportional TS-via number adjustment based on binary search and the full thermal resistive model will be called. Details of the multilevel routing and TS-via optimization algorithm are available in [7].

In [7] the TS-via number assigned to each tile is proportional to the temperature increase of that tile. We call such an approach *via planning proportional to temperature (VPPT)*. The rationale behind *VPPT* is intuitive: more TTS-vias should be placed at hotter regions. Nevertheless, they show that this simple heuristic, when embedded in the multilevel framework, can reduce the TS-via number by around 79% compared to the straightforward post-processing TTS-via insertion. However, our study in this paper shows that *VPPT* method can be greatly improved, and we can achieve additional 69% TS-via reduction compare to *VPPT*.

In this work we will use the similar multilevel framework with an improved TS-via planning algorithm — the alternating direction TS-via planning algorithm (*ADVP*) — at the initial solving and the refinement stages.

3. RESISTIVE THERMAL MODEL AND TTS-VIA PLANNING PROBLEM FORMULATION

TS-via planning will be carried out during the coarsest level and the following refinement levels. Usually, wirelength is usually more critical. Therefore, we only consider TTS-via number minimization in this work and assume a given STS-via assignment result at every planning level. However, since the TTS-vias will take up the routing resource, the following wire planning will be affected by the TTS-via insertion.

The TTS-via planning problem input includes the fixed positions of the cells/macros and the number of STS-vias at each tile. Only the TTS-vias are being planned. The TTS-vias are inserted into the whitespace between the cells/macros at the device layers.

Given a required temperature, the problem of TTS-via optimization is one of minimizing the total TTS-via number subject to the temperature constraint and TS-via capacity constraint.

3.1 Thermal Resistive Model

In this work we only consider the steady-state heat problem of 3-D IC. Also, we only consider the heat generated by transistor switches, so the macro blocks are treated as

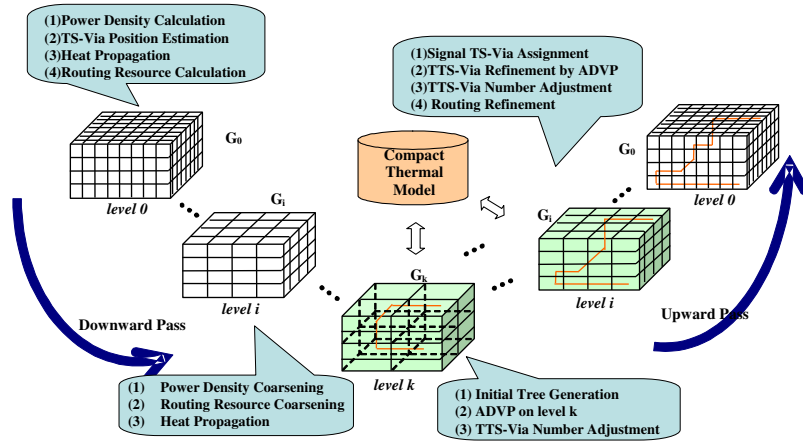


Figure 2: Multilevel Routing and TS-Via Planning Framework

the only heat sources with constant power densities. Since a heat sink is usually attached to the substrate, the bottom side of the tile stack is isothermal of constant room temperature, $27^{\circ}C$. The four side walls and top of the chip are treated as adiabatic, since the chip is usually packaged in thermally insulated materials.

There is a well-known duality between heat transfer and electrical current flow. Heat flow passes a thermal resistance like an electrical current going through a resistance. The temperature at any point is analogous to the voltage at that point; a heat source is analogous to a fixed current source, with the current value being the heat flow value. Therefore, one way to model temperature is to treat the chip as a circuit [3, 12, 20] of thermal resistances.

We chose to use the thermal resistive model proposed by Wilkerson, et al. in [20], which explicitly models the TS-vias. They proposed several versions of the resistive network models and we use the simplest version during the TS-via planning because of the compact size. The full resistive model is used during the final TS-via number adjustment and temperature verification. A tile structure is imposed on the circuit stack, as shown in Figure 3(a). Each tile stack contains an array of tiles, one from each device layer, as shown in Figure 3(b). A tile contains a certain number of TS-vias at the center. A voltage source is used for the isothermal base of room temperature, and current sources are present in every tile to represent heat sources.

A tile stack is modeled as a resistive network like the one shown in Figure 3(c). The tile stacks are connected by lateral resistances. There are two kinds of resistances in the circuits, the fixed and the variable. Fixed resistances include all lateral resistances and the bottom layer resistances in the tile stack, both shown as the gray resistances in Figure 3.

Variable resistances are shown as the black resistances in Figure 3. Given the TS-via number of every tile and the position of macro blocks, i.e., the heat sources, the circuit of thermal resistances can be solved by a linear solver, such as SPICE.

3.2 Preliminaries

Before presenting the details of the NLP problem formulation and the TTS-via planning algorithm, we will first introduce the symbols used in this paper. Suppose the design is divided into a 3-D array of tiles, $\{tile_{i,j,k} | 1 \leq i \leq X, 1 \leq j \leq Y, 1 \leq k \leq Z\}$.

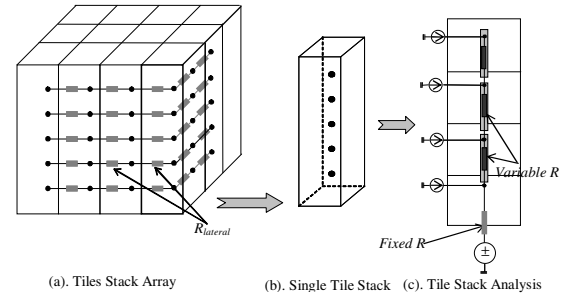


Figure 3: Compact Resistive Thermal Model

The inputs to our TTS-via planning problem include the following:

- T_0 : room temperature at the heat sink
- T_{input} : user required temperature
- $P_{i,j,k}$: power density at $tile_{i,j,k}$, which is determined by the macro blocks that overlap with $tile_{i,j,k}$
- $s_{i,j,k}$: STS-via number at $tile_{i,j,k}$, which is determined by the router
- $c_{i,j,k}$: TS-via capacity of $tile_{i,j,k}$

We use a commercial FEM-based thermal simulation tool, CFD-ACE+ [1][18] to compute a set of process technology related constraints, including:

- γ : thermal resistance of one TS-via, which is a technology-related constant
- $Rt_{i,j,k}$: thermal resistance of $tile_{i,j,k}$ without TS-vias, which is determined by the technology and the tile size
- $t_{i,j,k}$: equivalent TS-via number of a $tile_{i,j,k}$, $t_{i,j,k} = \gamma / Rt_{i,j,k}$. If we assume all tiles are of same shape and size, all device layers are fabricated using the same technology, then all $t_{i,j,k} = t$ will be the same.
- R_l : fixed lateral resistances between tiles

As long as the technology file is the same, those constants can be used in any design without running the thermal simulation again.

The variables in our problem are:

- $v_{i,j,k}$: temperature at $tile_{i,j,k}$
- $I_{i,j,k} \geq 0$: the vertical heat flow from $tile_{i,j,k+1}$ to $tile_{i,j,k}$
- $a_{i,j,k} \in Z^*$: total TS-via number at $tile_{i,j,k}$
- $Rv_{i,j,k}$: thermal resistance of all TS-vias in $tile_{i,j,k}$, $Rv_{i,j,k} = \gamma/a_{i,j,k}$
- $R_{i,j,k}$: the vertical thermal resistance between $tile_{i,j,k+1}$ and $tile_{i,j,k}$, which is the effective resistance of $Rt_{i,j,k}$ and $Rv_{i,j,k}$ connected in parallel

$$\begin{aligned}
R_{i,j,k} &= \frac{1}{1/Rt_{i,j,k+1} + 1/Rv_{i,j,k}} \\
&= \frac{t/\gamma + a_{i,j,k}/\gamma}{t + a_{i,j,k}}
\end{aligned} \quad (1)$$

3.3 An NLP Problem Formulation Based on Resistive Model

Under the resistive thermal model, the TS-via minimization problem under temperature constraint can be formulated as a constrained NLP problem **P**. Instead of $a_{i,j,k}$, V and I are used as variables in the formulation.

The total TS-via number, which is the objective function to be minimized, is then calculated as,

$$f(V, I) = \sum_{k \geq 2} a_{i,j,k} = \sum_{k \geq 2} \left(\frac{\gamma I_{i,j,k}}{v_{i,j,k} - v_{i,j,k-1}} - t \right) \quad (2)$$

Since it is difficult to handle integer constraints, $a_{i,j,k}$ is relaxed to a real number. Rounding will be necessary after solving the relaxed fractional TS-via planning problem. The constraints are listed as follows.

a. Temperature constraints.

$$T_0 \leq v_{i,j,k} \leq T_{input} \quad (3)$$

b. Tile capacity constraints. The amount of TS-vias assigned to each tile should not exceed the capacity of the tile.

$$\frac{\gamma I_{i,j,k}}{v_{i,j,k} - v_{i,j,k-1}} - t \leq c_{i,j,k} \quad (4)$$

c. Minimum TS-via number constraints. In order to avoid wirelength increase, $a_{i,j,k}$ should be larger than or equal to $s_{i,j,k}$, so that the STS-vias will not be moved to a position where detours will be introduced.

$$\frac{\gamma I_{i,j,k}}{v_{i,j,k} - v_{i,j,k-1}} - t \geq s_{i,j,k} \quad (5)$$

d. Kirchoff's current law (KCL). For each node j in V except the ground, the sum of incoming heat flows should be the same as the sum of outgoing heat flows. For node j , let $B(j)$ be the set of edges that connect with j , d_i be the direction of the heat flow on edge i . d_i is 1 when I_i is incoming and -1 otherwise.

$$\sum_{i \in B(j)} d_i I_i = 0 \quad (6)$$

TS-via minimization is to minimize (2) subject to equation (3) (4) (5) and (6). The problem is a constrained nonlinear optimization problem which is generally difficult to solve. We noticed that a similar problem formulation is used in

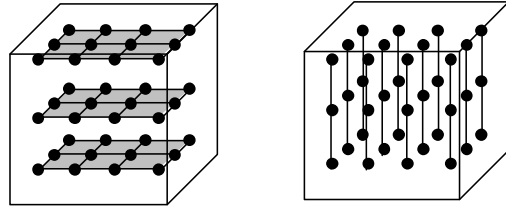


Figure 4: Alternating Direction Via Planning

research related to power/ground network optimization [5, 17]. However, the two-step relaxation (fixing I and fixing V) in these papers cannot be applied to our problem since there are fixed resistances in the thermal resistive network.

In order to efficiently solve the TS-via minimization problem, we also propose a two-step relaxation of the original problem. At step 1, we fix the (x, y) locations of the TS-vias and only move the TS-vias in the z direction, i.e., different layers for number minimization. At step 2, we fix the layers of the TS-vias and move them horizontally within each layer. The algorithm will iterate between step 1 and step 2 to search for a solution.

4. ALTERNATING DIRECTION TTS-VIA MINIMIZATION

We develop a multilevel alternating direction TTS-via planning algorithm (*m-ADVP*) based on implicit heat flow analysis instead of explicit thermal profiling. At each planning level, the alternating direction TTS-via planning algorithm, *ADVP*, iteratively alternates between vertical TTS-via distribution and horizontal TTS-via distribution.

4.1 ADVP Algorithm

The input of the *ADVP* algorithm is a 3-D array of tiles ($\Omega = M \times N \times Z$), each with information including power density, TS-via capacity, STS-via number, etc. Given an initial number of TTS-vias, A_0 , *ADVP* will assign them to each tile so that the total number of TS-vias can be minimized with the maximum temperature lower than T_{input} . At the initial solution generation step, Ω is the whole circuit, while during refinement, Ω is the part of the circuit covered by one tile stack at the coarser level.

It is difficult to simultaneously consider the heat flows in all x , y and z directions. Therefore, we separate the TTS-via planning into two steps: the vertical TTS-via planning and the horizontal TTS-via planning, as shown in Figure 4.

The vertical TTS-via planning distributes the TTS-vias to different device layers. The vertical TTS-via planning problem can be formulated as a convex programming problem, where an analytical optimal solution can be derived in some cases.

The horizontal TTS-via planning assigns TTS-vias within one device layer to different tiles. For horizontal TTS-via planning, we propose the *heat propagation* and the *path counting* techniques to get a heat flow estimation for the TTS-via distribution guidance.

4.1.1 Vertical TTS-via Distribution

If we replace the R_l inside Ω with short wires, and assume constant incoming heat flow, Ω can be modeled as a chain of resistors R_k , as shown in Figure 5. R_k is an effective resistor of $\{R_{i,j,k} | i = 1, \dots, M, j = 1, \dots, N\}$ connected together

in parallel. This assumption is reasonable for technologies where the macro blocks are located at silicon layers, whose thermal conductivity is good. For the silicon-on-insulator (SOI) type of technology, however, the blocks are buried in insulator materials with very low thermal conductivity and each small tile stack should be considered independently.

Let $a_k = \sum_{1 \leq i \leq M, 1 \leq j \leq N} (a_{i,j,k} + t)$ be the total TS-via number in layer k of Ω , then R_k can be calculated as follows.

$$R_k = \frac{\gamma}{\sum_{1 \leq i \leq M, 1 \leq j \leq N} (a_{i,j,k} + t)} = \frac{\gamma}{a_k} \quad (7)$$

In this way, the circuit can be modeled as the resistive chain shown in Figure 5. The temperature of the nodes in such a chain can be calculated through an Elmore delay-like equation [4].

$$v_k = \sum_{i=1}^k R_i \sum_{j=i}^Z P'_j + T_0 \quad (8)$$

Let $s_k = \sum_{1 \leq i \leq M, 1 \leq j \leq N} s_{i,j,k}$ and $c_k = \sum_{1 \leq i \leq M, 1 \leq j \leq N} c_{i,j,k}$. The temperature-constrained TS-via optimization problem on the chain model can be written as follows.

$$\begin{aligned} \min \quad & \sum_{k=2}^Z a_k \\ \text{s.t.} \quad & v_Z = \sum_{k=1}^Z R_k \sum_{l=k}^Z P'_l + T_0 \\ & = \sum_{k=1}^Z \frac{\gamma}{a_k} \sum_{l=k}^Z P'_l + R_b \sum_{k=1}^Z P'_k + T_0 \\ & \leq T_{input} \\ & s_k \leq a_k \leq c_k, \quad k = 2, \dots, Z \end{aligned} \quad (9)$$

The problem is a convex programming problem and can be solved optimally by any convex programming package. If we take away the second set of constraints, which means the capacity is sufficient and there is no STS-via assigned, the problem can be directly solved through the *KKT* optimality condition. Let

$$g = \sum_{j=2}^Z a_j + \nu \left(\sum_{j=2}^Z \frac{K_j}{a_j} + T_0 - T_{input} \right) \quad (10)$$

where $K_j = \gamma \sum_{k=j}^Z P'_k$ is a constant. The *KKT* condition requires

$$\frac{\partial g}{\partial a_k} = 0, \quad 2 \leq k \leq Z \quad (11)$$

Therefore, the optimal solution from solving (11) will require that the ratio of the TS-vias assigned to different device layers satisfies the following equation.

Theorem: For a resistive chain, if there is no constraint of the TS via number assigned to each tile, the solution to the temperature constrained TS-via minimization problem satisfies the following:

$$\begin{aligned} a_Z : a_{Z-1} : \dots : a_3 : a_2 \\ = \sqrt{P'_Z} : \sqrt{P'_Z + P'_{Z-1}} : \dots : \sqrt{\sum_{k=3}^Z P'_k} : \sqrt{\sum_{k=2}^Z P'_k} \end{aligned} \quad (12)$$

For efficiency reasons, in our implementation we directly use the equation in (12) for vertical TS-via distribution and force the resulting a_k into the constraint range $[s_k, c_k]$. From our experiments, the quality of results is similar because in most cases the range is fairly loose.

Please note the *VPPT* approach in [7] assumes that the ratio total TS-via number at each layer as

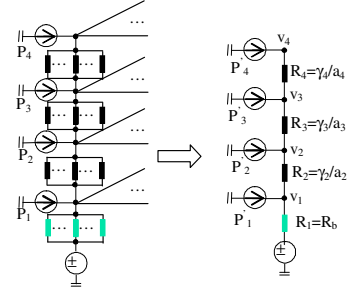


Figure 5: Vertical TS-Via Planning Models

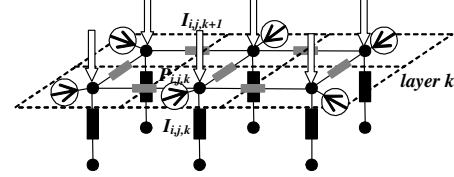


Figure 6: Horizontal TS-Via Planning Models

$$\begin{aligned} a_Z : a_{Z-1} : \dots : a_3 : a_2 \\ = \Delta v_Z : \Delta v_{Z-1} : \dots : \Delta v_3 : \Delta v_2 \\ = P'_Z : (P'_Z + P'_{Z-1}) : \dots : (\sum_{k=3}^Z P'_k) : (\sum_{k=2}^Z P'_k) \end{aligned} \quad (13)$$

Compared to equation (12), it is not difficult to conclude that *VPPT* cannot generate the optimal results for vertical TS-via distribution.

4.1.2 Horizontal TTS-Via Distribution

After assigning TTS-vias to different layers within Ω , *ADVP* will distribute the TTS-vias within every layer k in Ω . The equivalent thermal resistive network of the intra-layer TTS-via distribution is shown in Figure 6. However, even in such a simplified resistive network, the TTS-via minimization problem is still a constrained nonlinear programming problem and difficult to solve. Also, it is not easy to compute the target temperature at each device layer. Therefore, within each layer, we further simplify the problem by assuming a given total TTS-via number A_k and an objective of even temperature increase so that “hot spots” can be avoided.

However, TS-vias cannot be placed directly at the hottest spots since those places are occupied by macro blocks or cells. Instead of inserting TTS-vias right on the hot spots, sometimes we can only put TTS-vias around the hot spots. Figure 7 is a temperature map of a 3-D circuit ami33, bottom layer, generated by CFD-ACE+. The two hottest places (on the right-hand side), shown with dark colors, are blocked by macros. The temperature at the empty channels around the hot blocks, however, is lower than the actual hot spot temperature and cannot attract enough TS-vias. Therefore, when applied to horizontal TS-via distribution, *VPPT* cannot handle cases with large macro blocks.

In order to solve the problem, we distribute the TS-vias according to the vertical heat flow after proper *heat propagation*. We take an initial even TS-via distribution, where

$$a_{i,j,k} = A_k \cdot c_{i,j,k} / \sum_{1 \leq i \leq M, 1 \leq j \leq N} c_{i,j,k} \quad (14)$$

Then thermal resistance at the whitespace is the same everywhere and much lower than the thermal resistance at the

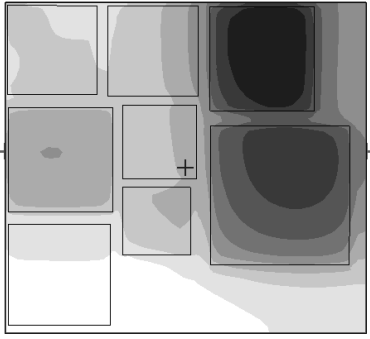


Figure 7: An Example Temperature Map of a Device Layer with Eight Blocks

blocked area. The heat generated by the “hot blocks” will then flow to the neighboring whitespace with TS-vias because the thermal resistance there is much lower than the center of the hot blocks. We then distribute the TS-vias according to the vertical heat flow at each tile, $I_{i,j,k}$.

$$a_{i,j,k} = A_k \cdot \frac{I_{i,j,k}}{\sum_{1 \leq i \leq M, 1 \leq j \leq N} I_{i,j,k}} \quad (15)$$

In this way, the heat generated at the center of the macro blocks, where there is no space for TS-via insertion, is “propagated” to the boundary tiles and the tiles below them. The heat flow of layer k , $I_{i,j,k}$, depends on the heat flow of the upper layer, $I_{i,j,k+1}$. Therefore, our horizontal TTS-via assignment algorithm starts from the top layer and ends at bottom layer.

4.1.3 Heat Flow Computation

The heat flow $I_{i,j,k}$ can be calculated by the thermal resistive model, which is computationally expensive since the $I_{i,j,k}$ value needs to be updated frequently. In order to speedup heat propagation calculation, we calculate the heat flow through *path counting*. To simplify the calculation, we also assume the temperature at the lower layer is uniform. The assumption is reasonable with our even temperature increase objective of intra-layer TTS-via assignment.

In the equivalent circuit, every tile $tile_{i,j,k}$ has a total incoming heat flow of

$$H_{i,j,k} = I_{i,j,k+1} + P_{i,j,k} \quad (16)$$

where $I_{i,j,k+1}$ is the heat flow from the above and $P_{i,j,k}$ is the heat flow generated by the heat source located at $tile_{i,j,k}$.

Since the heat sink is located at the bottom of the circuit stack, heat will generally flow to the lower layers. In the resistive network, there are many dissipating paths from a layer k tile to the tiles at layer $k-1$. In our calculation, we will only consider the M shortest paths. In our experiments we set M as 10. However, our results show that larger numbers will not improve the final results significantly.

The total thermal resistance of each path is calculated by adding the resistances on the path together. The heat flow going through each path is inversely proportional to the total resistance of that path.

Let us look at the example in Figure 8. Assume we count the five shortest paths p_1, \dots, p_5 , from $tile_{i,j,k}$ to layer $k-1$.

$$\begin{aligned} R(p_1) &= R_l \cdot distance + R_{i-1,j+1,k} \\ &= R_l \cdot distance + \gamma / (a_{i-1,j+1,k} + t) \end{aligned} \quad (17)$$

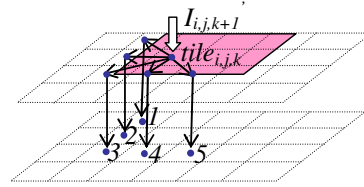


Figure 8: Heat Dissipating Paths to the Lower Layer ($M=5$)

Table 1: *ADVP* Algorithm

<i>Input:</i> $\Omega = M \times N \times Z$, $I_{i,j,k}$, $A_0[k]$ from the previous planning result
<i>Output:</i> assigned TS-via number for each tile $a_{i,j,k}$
for every device layer k , starting from the top
{
set initial distribution = even distribution of $A_0[k]$,
heat propagation for layer k to update $\{I_{i,j,k}\}$
}
while not converged
{
for each i, j , $1 \leq i \leq M$, $1 \leq j \leq N$,
vertical TS-via distribution for $\{tile_{i,j,k} 1 \leq k \leq Z\}$
for every device layer k , starting from the top
{
horizontal TS-via distribution for
$\{tile_{i,j,k} 1 \leq i \leq M, 1 \leq j \leq N\}$
heat propagation for layer k to update $\{I_{i,j,k}\}$
}
}

The heat flow on p_1 is then calculated as

$$I(p_1) = H_{i,j,k} \frac{1/R(p_1)}{\sum_{j=1}^5 1/R(p_j)} \quad (18)$$

After path counting for every tile, we can calculate $I_{i,j,k}$, where $I_{i,j,k}$ is the sum of the heat flow on all paths ending at $tile_{i,j,k-1}$. Then, TS-vias will be assigned to tiles proportional to $I_{i,j,k}$.

After horizontal TS-via distribution at each layer, the heat flow map will be updated and used by the following vertical TS-via distribution as well. The vertical TS-via distribution equation can be rewritten using the heat flow values.

$$\begin{aligned} a_Z : a_{Z-1} : \dots : a_3 : a_2 \\ = \sqrt{I_Z} : \sqrt{I_{Z-1}} : \dots : \sqrt{I_3} : \sqrt{I_2} \end{aligned} \quad (19)$$

where $I_k = \sum_{1 \leq i \leq M, 1 \leq j \leq N} I_{i,j,k}$.

In the multilevel framework, *path counting* and *heat propagation* are first performed during coarsening with an initial even TS-via distribution. The propagated heat information based on the estimated TS-via number is also coarsened during the coarsening process. During initial solving and refinement stages, after the horizontal TTS-via assignment process finishes each layer, the actual TS-via number is used for another round of path counting and heat propagation to generate a more accurate heat flow map. In summary, the *ADVP* algorithm for Ω is shown in Table 1. When *ADVP* is applied to the coarsest and the refinement levels in a multilevel framework, we call it a *m-ADVP* algorithm.

5. EXPERIMENTAL RESULTS

We implemented the multilevel *ADVP* algorithm (*m-ADVP*), multilevel *VPPT* algorithm (*m-VPPT*), and the even TS-via distribution scheme (*EVEN*) in Linux using C++, and

Table 2: 3-D Routing Examples

circuits	#nets	init T (°C)	flat level #tile	#STS-via
ami33	133	157.9	22×22×4	500
ami49	407	191.8	60×59×4	889
n100	884	208.1	44×40×4	1510
n200	1584	195.7	42×40×4	2744
n300	1892	190.2	50×60×4	3559

Table 3: Comparison of m -ADVP and Solving NLP

circuits	m -ADVP			solving P		
	T (°C)	TS -via #	planning time(s)	T (°C)	TS -via #	planning time(s)
ami33	77.0	1282	1.55	77.0	1192	942.2
ami49	77.0	20956	13.5	77.2	21138	1850
n100	77.0	11887	7.66	77.2	11707	874.4
n200	77.0	13980	12.24	77.2	13961	799.9
n300	77.0	17646	20.44	77.0	18044	1583.6
Avg.		1.0	1.0		0.99	200.4

tested the three schemes on Xeon 2Ghz machine. The TS-via planning algorithm is integrated with a multilevel routing framework. At every refinement level, STS-vias are assigned for wirelength minimization before the TTS-via assignment.

Table 2 shows the total multi-pin net number and initial temperature of the circuits. A four-device layer configuration, is assumed for all circuits. Each device layer is silicon based, and there are two metal routing layers on top of each device layer. A thermal TS-via will extend to the metal layers above and below its device layer. Each block is randomly assigned with a power density value between $10^5(w/m^2)$ and $10^7(w/m^2)$ [19]. The floorplan layout is generated by a 3-D thermal-driven floorplanning tool [6] with whitespace reserved between blocks for inter-layer connections. The global routing result is generated by a 3-D wirelength-driven routing algorithm [7]. The temperatures of the circuits after global routing with no thermal TS-via insertion are also listed.

We used a conjugate gradient penalty function-based NLP solver package [15] in the multilevel framework. The problem formulation based on the resistive model in Section 3.3 (**P**) is solved at every refinement level with an initial solution provided by *ADVP* and proper TS-via number adjustment. The final temperature is calculated by the accurate resistive model. Table 3 shows the result of solving **P**. We can see that for the some circuits, the results can be improved, but with much longer runtime ($200\times$ longer).

Table 4 shows the results of the four different TS-via planning schemes: m -ADVP, the multilevel *VPPT*, f -ADVP, the flat *ADVP*, (m -*VPPT*) approach in [7], and a simple approach of even TS-via distribution (*EVEN*). All schemes are required to bring the temperature down to 77°C (350 in absolute temperature). The final temperatures are calculated by the accurate full resistive thermal model in [20] at the finest level. Figure 9 shows the final temperature distribution for the top device layer of example ami33. For the same temperature constraint, m -ADVP can reduce the total TS-via number by 11% over the flat version, by 68% over m -*VPPT*, and $3.55\times$ over *EVEN*. The “area ratio” columns show the percentage of the chip area occupied by TS-vias.

The complete routing flow is finished by further wire planning and a grid-based detailed router. The results are shown in Table 5. We can see that the completion rate of a circuit is affected by the number of the TS-vias assigned to it. Therefore, under the m -*VPPT* scheme, the router can reach

Table 5: Final Routing Results

circuits	m -ADVP		m -VPPT [7]		<i>EVEN</i>	
	comp. rate	r.t.(s)	comp. rate	r.t.(s)	comp. rate	r.t.(s)
ami33	100%	7.38	100%	6.92	100%	7.45
ami49	97.2%	954.2	94.4%	1173.8	79.4%	1835.3
n100	94.3%	4136.9	88.3%	6210.7	61.6%	15161.8
n200	94.0%	14922.5	94.0%	13340.5	68.2%	46352
n300	99.1%	1431	91.7%	4110.8	58.0%	13618.1
Avg.	96.9%	1.0	93.7%	1.49	73.44%	3.84

the highest completion rate of 96.9%. Also, when a design is hard to route, the router will take a longer runtime to search in a larger space for a solution. To further improve the completion rates, we can also consider the congestion during the TS-via planning in addition to the TS-via minimization.

6. CONCLUSIONS

We formulate the TTS-via minimization problem as a constrained NLP and solve it by solving a sequence of simplified subproblems in alternating direction a multilevel framework. We propose an efficient and effective multilevel alternating direction TTS-via planning algorithm (m -ADVP) with vertical TTS-via distribution solved by convex programming and horizontal TTS-via distribution which is solved by *path counting* and *heat propagation*. Our algorithm is based on implicit heat flow analysis instead of explicit thermal profiling. Thermal modeling and optimization are combined in one multilevel framework. Experimental results show that our algorithm can reduce 68% of TS-vias over the m -*VPPT* [7] method with similar runtime. We also solve the NLP directly through an off-the-shelf solver. Compared to directly solving, m -ADVP achieves $200\times$ speedup with similar solution quality.

As we discussed before, the algorithm proposed in this paper is focused on TTS-via planning. The position of STS-vias can also be optimized for both temperature and wirelength. Also, TTS-via planning can consider the congestion information in addition to temperature to assist routing. The multi-objective via planning problems will be more difficult.

TS-via planning can also be used during the floorplanning and placement process, or used as a stand-alone process after floorplanning or placement, to determine the whitespace assignment between the macro blocks or cells. Considering TS-vias will provide more accurate temperature estimation for temperature optimization during floorplanning or placement.

7. REFERENCES

- [1] *CFD-ACE+ Module Manual*, 2002.
- [2] K. Banerjee, S. Souri, P. Kapur, and K. Saraswat. 3-D ICs: A Novel Chip Design for Improving Deep-Submicrometer Interconnect Performance and Systems-on-Chip Integration. *Proc. of the IEEE*, 89(5):602–633, May 2001.
- [3] T.-Y. Chiang, K. Banerjee, and K. C. Saraswat. Compact Modeling and SPICE-Based Simulation for Electrothermal Analysis of Multilevel ULSI Interconnects. *Proc. IEEE International Conference on Computer-Aided Design*, pages 165–172, 2001.
- [4] T.-Y. Chiang, S. J. Souri, C. O. Chui, and K. C. Saraswat. Thermal Analysis of Heterogeneous 3-D ICs

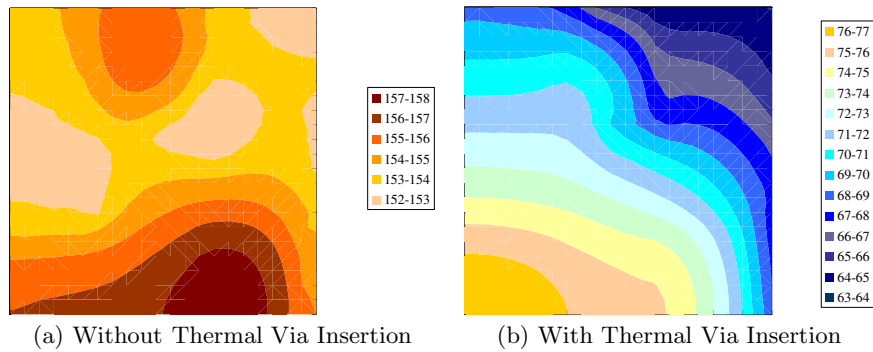


Figure 9: Temperature Distribution of the Top Layer, ami33

Table 4: Comparison of Different TS-Via Insertion Approaches

circuits	<i>m</i> -ADVP				<i>f</i> -ADVP				<i>m</i> -VPPT [7]				<i>EVEN</i>			
	T (°C)	TS -via #	area ratio	<i>r.t</i> (s)	T (°C)	TS -via #	area ratio	<i>r.t</i> (s)	T (°C)	TS -via #	area ratio	<i>r.t</i> (s)	T (°C)	TS -via #	area ratio	<i>r.t</i> (s)
ami33	77.0	1282	2.5%	1.55	77.0	1415	2.7%	1.46	77.1	1801	3.5%	1.76	77.1	2315	4.5%	1.62
ami49	77.0	20956	0.9%	13.5	77.0	20182	0.8%	15.05	77.1	43794	1.8%	12.15	76.9	166366	6.8%	16.17
n100	77.0	11887	1.5%	7.66	77.8	14617	1.9%	6.9	77.0	22211	2.8%	8.31	76.8	30853	3.9%	7.54
n200	77.0	13980	1.8%	12.24	77.2	15236	2.0%	11.6	77.2	18835	2.4%	10.89	77.1	30346	3.9%	12.21
n300	77.0	17646	1.3%	20.44	77.1	20154	1.5%	20.55	77.1	30161	2.2%	21.73	76.9	57342	4.2%	22.42
Avg.		1.0	1.6%	1.0		1.11	1.8%	0.98		1.68	2.6%	1.01		3.55	4.7%	1.06

with Various Integration Scenarios. *IEEE International Electron Devices Meeting (IEDM) 2001 Technical Digest*, pages 681–684, Dec. 2001.

- [5] S. Chowdhury. Optimum design of reliable IC power networks having general graph topologies. *Proc. 26th ACM/IEEE Design Automation Conf.*, pages 787–790, 1989.
- [6] J. Cong, J. Wei, and Y. Zhang. A Thermal-Driven Floorplanning Algorithm for 3D ICs. *Proc. IEEE International Conference on Computer-Aided Design*, 2004.
- [7] J. Cong and Y. Zhang. Thermal-Driven Multilevel Routing for 3-D ICs. *Proc. of the Asia South Pacific Design Automation Conference*, pages 121–126, Jan. 2005.
- [8] S. Das. *Design Automation and Analysis of Three-Dimensional Integrated Circuits*. PhD thesis, Massachusetts Institute of Technology, May 2004.
- [9] B. Goplen and S. Sapatnekar. Efficient Thermal Placement of Standard Cells in 3D ICs using a Force Directed Approach. *Proc. IEEE International Conference on Computer-Aided Design*, 2003.
- [10] B. Goplen and S. Sapatnekar. Thermal Via Placement in 3D ICs. *Proc. of International Symposium on Physical Design*, pages 167–174, Apr. 2005.
- [11] S. B. Horn. Vertically integrated sensor arrays VISA (Invited). *Defense and Security Symposium*, 2004.
- [12] W. Huang, M. R. Stan, K. Skadron, K. Sankaranarayanan, S. Ghosh, and S. Velusamy. Compact Thermal Modeling for Temperature-Aware Design. *Proc. ACM/IEEE 41st Design Automation Conference*, pages 878–883, 2004.
- [13] S. Im and K. Banerjee. Full Chip Thermal Analysis of Planar (2-D) and Vertically Integrated (3-D) High Performance ICs. *Technical Digest IEEE International Electron Devices Meeting (IEDM)*, pages 727–730, Dec. 2000.
- [14] S. Lee, T. Lemczyk, and M. Yovanovich. Analysis of Thermal Vias in High Density Interconnect Technology. *Proc. 8th IEEE Semi-Therm Symposium*, pages 55–61, Feb. 1992.
- [15] W. Naylor and B. Chapman. <http://www.willnaylor.com/wilib.html>.
- [16] A. Raman, M. Turowski, and M. Mar. Layout-based Full Chip Thermal Simulations of Stacked 3D Integrated Circuits. *International Mechanical Engineering Congress and R&D Expo*, Nov. 2003.
- [17] X.-D. Tan, C.-J. Shi, and J.-C. Lee. Reliability-Constrained Area Optimization of VLSI Power/Ground Networks Via Sequence of Linear Programmings. *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, 22(12):1678–1684, Dec. 2003.
- [18] Z. Tan, M. Furmanczyk, M. Turowski, and A. Przekwas. CFD-Micromesh: A Fast Geometrical Modeling and Mesh Generation Tool for 3D Microsystem Simulations. *Int. Conf. MSM 2000*, pages 712–715, Mar. 2000.
- [19] C. Tsai and S. Kang. Cell-Level Placement for Improving Substrate Thermal Distribution. *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 19(2):253–266, Feb. 2000.
- [20] P. Wilkerson, M. Furmanczyk, and M. Turowski. Compact Thermal Modeling Analysis for 3D Integrated Circuits. *11th International Conference Mixed Design of Integrated Circuits and Systems, Szczecin, Poland*, Jun. 2004.