Thermal and Wirelength Optimization With TSV Assignment for 3-D-IC

Yi Zhao, Cong Hao, *Member, IEEE*, and Takeshi Yoshimura, *Member, IEEE*

***Abstract*— 3-D-IC is an emerging technology that overcomes the drawbacks of 2-D-IC and a vital structure called through-silicon via (TSV) is used to connect the adjacent layers vertically. In this paper, based on graph theories, we formulate the TSV assignment as an integer multicommodity min-cost (IMCMC) problem, and using multilevel algorithm, we get optimal solution. During the assignment, the capacity of the grid could be insufficient to contain the flows so that conditional grid extension effectively relieves the capacity limits and accelerate running speed. Thermal resistive model is introduced to simulate the heat dissipation of chip and estimate temperature distribution each layer. By inserting the thermal TSV, the temperature of superheated region is reduced. Moreover, considering about a more complicated situation, we propose a method to assign multipins in a net. The experimental results demonstrate our model in the IMCMC network using multilevel algorithm achieve optimized congestion, less wirelength, higher running speed, and reduced temperature.**

***Index Terms*— 3-D-IC, thermal through silicon-via (TSV), TSV assignment.**

I. Introduction

THE invention and challenges of 3-D-ICs are described in the following.

The Invention of 3-D-ICs: In the latest decade, an emerging technology named 3-D integrated circuits (3-D-IC) supported by through-silicon via (TSV) comes up, which greatly extends the space of IC development together with supplementing the drawbacks of 2-D-IC. The basic structure of a 3-D-IC fabricated in via-middle is shown in Fig. 1.

The components (gate arrays) are located on the different layers of chip dies, communication wires are crisscross in the chip. Chips with different functions pile layer by layer in the vertical directions. There are bulks whose radius is larger than ordinary vias called TSVs, taking the responsibility of connecting and communicating between dies. Fig. 2 shows an entire structure of complicated 3-D-IC.

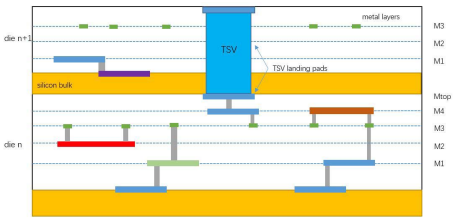


Fig. 1. TSVs, TSV landing pads, and connections to TSV landing pads.

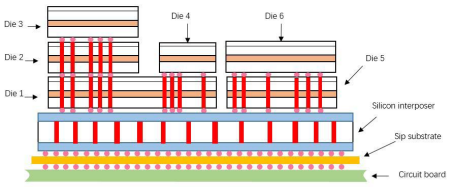


Fig. 2. Entire 3-D-IC module.

The Challenges of 3-D-ICs: However, the complex structure makes trouble to the fabrication, such as microchannels. The possibility of mistakes inevitably becomes larger and the cost increases. The heat dissipation of the chip is also a vital factor in the IC design. This is because up to millions of components produce a great quantity of heat in such a tiny room, the temperature may sharply increase to make danger and deteriorate the performance. At this point, cooling method has to be made use of and thermal TSV is effective [2].

A. Related Research

Many previous dissertations [1]–[13] have already been proposed for TSV assignment of 3-D-IC. Current research result is mainly divided into two aspects, one is focusing on multilevel flow in order to optimize the wirelength and congestion, and the other is on thermal dissipation to improve the performance. On the one hand, the assignment of wire and TSV is a significant task because of the great number of components and microscaled chip area. The work in [4] paid attention to wirelength optimization in the multilevel. Reference [5] designed an innovative method to solve the multilevel problem by calculating the residual path in the flow network. The work in [6] proposed an integer multiflow min-cost flow algorithm to assign the wire in a coarsening-uncoarsening method level by level. In [8], in order to make

Manuscript received September 6, 2018; revised October 6, 2018; accepted October 10, 2018. Date of publication November 5, 2018; date of current version December 24, 2018. The review of this paper was arranged by Editor M. S. Bakir. *(Corresponding author: Yi Zhao.)*

Y. Zhao is with the Graduate School of Information, Production and System, Waseda University, Kitakyushu 808-0135, Japan, and also with the School of Electronic Science and Engineering, Southeast University, Nanjing 210018, China (e-mail : scchfzy@126.com).

C. Hao is with the Graduate School of Information, Production and System, Waseda University, Kitakyushu 808-0135, Japan, and also with the Institute of Automation, Shanghai Jiao Tong University , Shanghai 200240, China (e-mail: hc.onioncc@gmail.com).

T. Yoshimura is with the Graduate School of Information, Production and System, Waseda University, Kitakyushu 808-0135, Japan (e-mail: t-yoshimura@waseda.jp).

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

Digital Object Identifier 10.1109/TED.2018.2875933

signal TSV planning more flexible, TSV is integrated with pin assignment. The work in [10] reused 2-D blocks to construct 3-D rectilinear Steiner tree (RST) together with using a postfloorplanning TSV insertion method. What's more, fast lookup table estimation (FLUTE) algorithm in [11] was also referred to estimate the TSV locations.

On the other hand, the resistive thermal network in [2] is the most commonly used thermal increase model. However, in the calculation, matrix equations need solving, [1] proposed compression storage and left-looking LU (lower and upper triangular matrix) decomposition method to reduce the complexity of calculation, which greatly save the storage space and time.

Koyanagi *et al.* [12] and Bakir *et al.* [13] proposed many constructive ideas on the material and fabrication of 3-D-ICs.

B. Major Contribution

This paper adopts the *integer multicommodity min-cost (IMCMC) network* to search for the minimum wirelength. Compared with former works, multilevel chip is discussed. It is able to take the capacity and area into consideration precisely. It provides a flexible and convenient method to assign the components and wires using flow algorithms. In this way, we could discuss the assignment of all the nets and layers simultaneously and ensure the wirelength relatively the shortest. A *thermal increase model* is introduced to estimate the temperature increase. Combined with component properties, we propose a concrete thermal reduction algorithm. Other modifications are made to make the resolution satisfy the realistic circuits more. We summarize the contributions of this work as follows.

1) We propose a *conditional grid extension* method to extend the adjacent region of the effective grids in the uncoarsening step. In this way, the capacity of the region increases so that more wires could locate in the region.
2) Based on the *thermal resistive model*, we proposed a thermal optimization method. Types of components influence the area and power consumption. We regulated a method to decide the position and number to insert thermal TSVs. Approximate temperature increase caused by layer assignment between adjacent layers and heat reduction could be calculated.
3) We consider the condition of *multipins in a net*. We make use of 3-D RST supported by FLUTE algorithm.

The rest of this paper is organized as follows. Section II gives the problem formulation. Section III introduces the IMCMC network and the conditional grid extension method we propose. Then, we give the detail about the thermal increase model in Section IV. The algorithm of multipins in a net is in Section V. The experimental results are shown in Section VI. Finally, Section VII gives the consequences.

II. Problem Formulation

A. Problem Description

In a 3-D-IC module, the input is given as follows.

1) A 3-D-IC with n chip dies D=$\{d_1, d_2, \ldots, d_n\}$ and $n-1$ TSV layers L=$\{l_1, l_2, \ldots, l_{n-1}\}$, where l_k TSV layer connects d_k and d_{k+1} chip dies.
2) Each chip die is divided into many grids and each grid has a certain capacity c_g defining the number of components could be allocated to. $\forall c_g \geqslant 0$.
3) The 3-D netlist includes m nets which describes the situation of connections. Each net is denoted as a two-pin net composed of $\eta_k = s_k, t_k$. s is the source pin which is transferred from multipin on the toplayer by minimum spanning tree (MST), and t is the sink pin [14].
4) Every source pin s and sink pin t locates on a certain grid i associated with a position, denoted as $p_i = (x_i, y_i, z_i)$, $p_s = (x_s, y_s, z_s)$, $p_t = (x_t, y_t, z_t)$, where z is the number of chip and $1 \leqslant z_s < z_t \leqslant n$.
5) In the thermal increase model, the device property including the number and area of components, thermal resistance, and power dissipation of each layer is also given.

The source pin s and sink pin t of one net $\eta_k = (s, t)$ construct a bounding box. If we project them into one layer, we will get a 2-D rectangular minimum region. The layer that net passes has at most one TSV and the vertical distance between layers is constant so that we could estimate the wirelength by *Manhattan distance* [18]

$$wl(\eta) = d_{s,\tau_1} + \sum_{1 \leqslant i \leqslant z_t - z_s} d_{\tau_i, \tau_{i+1}} + d_{\tau_{z_t - z_s}, t} \tag{1}$$

where $d_{p,q}$ is the distance between the two adjacent layer p and q that calculated by Manhattan distance. s is the source pin, t is the terminal pin, τ is the layers net passes through. In this way, we get the estimation of wirelength by summing up every wirelength between layers and the vertical distance is ignored because it is constant.

The problem has a *constraint* for each grid i, the number of TSVs are contained must not exceed the capacity g_i.

The *purpose* of the work is to minimize the sum of wirelength of all of the nets $\eta_i = (s_i, t_i)$ during the TSV assignment

$$\min : \sum_{1 \leqslant i \leqslant m} wl(\eta_i). \tag{2}$$

Here, we may encounter a tradeoff between time and wirelength. We sacrifice time in steps like coarsening and uncoarsening, grid extension, building RST, insertion of thermal TSV, which result from iteration and additional process, so that we are able to get wirelength as short as possible. However, we also design to avoid too much excess time consumption so conditional grid extension is proposed.

B. IMCMC Network Description

We formulate the TSV assignment problem as an IMCMC network flow problem [14]. Given $G = (V, E)$, associated with costs and capacities on each edge. The vertices composed of $V = V_s \cup V_t \cup V_g$ and $g_{mn} \subset V_g$ which represents nth grid on mth layer. Each vertex g_{mn} has a capacity c_g. The edges $E = \{e | e = (u, v)\}$, where $u \subset V_s, v \subset V_g$, or $u \subset V_g$,

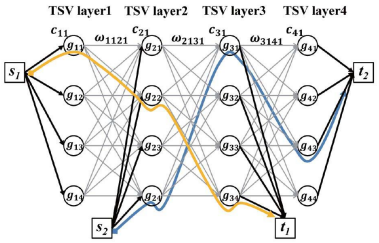


Fig. 3. Instance of TSV assignment of two nets on the IMCMC network.

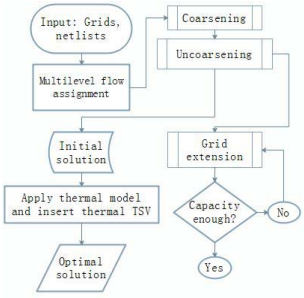


Fig. 4. Algorithm flow of multilevel assignment.

$v \subset V_t$, or $u, v \subset V_g$ in the direction from u to v. Edges have their own costs ω_e defined as the routing costs between the pins and TSVs, which is the same as Manhattan distance in (1). When assigning a net in the network, it is equivalent to defining a signal path from s to t and the flow on an edge (u, v) is denoted as $f_{u,v}^{s,t}$. The solution is to find the min-cost flow of the graph [6]. Fig. 3 sets an example about IMCMC network with two nets $\eta_1 = (s_1, t_1)$, $\eta_2 = (s_2, t_2)$ in yellow and blue, respectively. We found that the flow from s to t could determine the assignment solution of nets. Net1 passes through g_{11}, g_{22}, and g_{34} in three layers and correspondingly three TSVs are assigned to Net1.

Fig. 4 shows our multilevel assignment flow. The input contains grids and nets with multipins. Generating the IMCMC network, we manage the coarsening and uncoarsening steps to get the initial solution meanwhile optimizing by conditional grid extension. By applying the thermal module and inserting thermal TSVs, we estimate the heat distribution and reduce the temperature.

III. Multilevel Flow Assignment

In the multilevel problem, the most intractable point is to optimize the wire connection in the most efficient method. Considering the problem, we propose *coarsening and uncoarsening algorithm* in the IMCMC network to accomplish the target of reducing the number of edges in the calculation. Here, the essence of coarsening is division and grouping, uncoarsening is the inverse process. For a chip composed of $k \times k$ grids denoted as level ϵ, we make the adjacent grids into a group denoted as $\text{grd}_{ij}^{\epsilon}$. After coarsening for one time, we get $(k/2) \times (k/2)$ grids, that is to say, in the coarsening level ϵ the grids become $(k/2^{\epsilon-1}) \times (k/2^{\epsilon-1})$ until 2×2.

In the flow assignment, we define the grids have positive capacity and could contain TSVs as *effective grids*, effective, and ineffective grids are both included in the coarsened

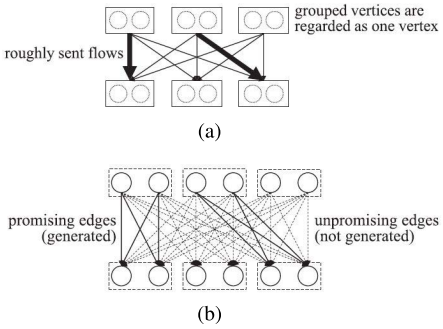


Fig. 5. Multilevel algorithm. (a) Flows are roughly sent on grouped vertices. (b) Only promising edges are generated from rough flows.

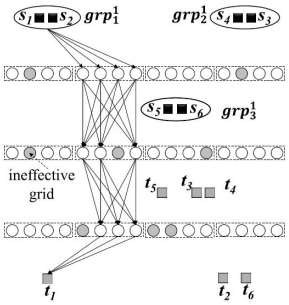


Fig. 6. Edges generating in coarsening graph.

graph but only effective grids could generate edges. In the assignment, the capacity might gradually decrease to 0 and the grids become ineffective. In the IMCMC network, only a small part of edges are actually used. Therefore, we define *promising edges* that have higher possibilities to be used in the solution. Only the promising edges could be generated between effective grids in the uncoarsening step and the unpromising edges are not generated. We send *rough flow* between the vertices and decide the effective grids in which TSV is inserted in Fig. 5(a). The rough flow is an initial flow sent to the network and would be adjusted to optimized result iteration by iteration. Then, promising edges are generated in the next uncoarsening step in Fig. 5(b) but the unpromising ones are not. Thus, the number of edges reduces. As Fig. 6 shows, promising edges are generated for each effective grid barring the ineffective ones (gray balls).

A. Conditional Grid Extension

In this solution, the number of effective grids during uncoarsening is usually limited, we cannot find a proper grid to insert TSVs in the adjacent regions, which results in attempts to other grids for more times. Therefore, we propose a *conditional grid extension* to extend the inspective region to get enough effective grids. Denoting the *extending radius* as r, the edges are generated for the effective grids in the region of radius r.

Related research was already done in [6], the residual flow will be assigned to other effective grids in the extended region if the capacity in the initial region is not sufficient. The extension is conducted in an iterative way so that the value of r shows magnificent impact to results. If the r was too small, the effective grids would be insufficient. If r was too large,

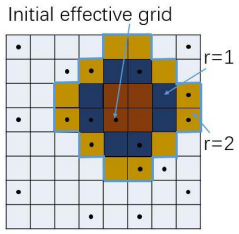


Fig. 7. Conditional grid extension.

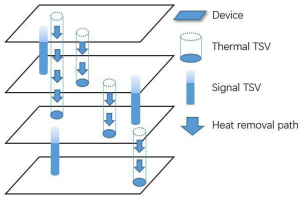


Fig. 8. Heat dissipation path.

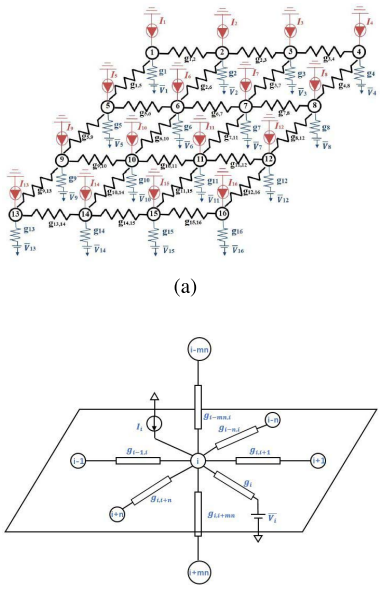
excessive resource (storage and time) would be sacrificed. The value of r is set to be conditional, which means the extending extent differs due to the different number of grids each level. Therefore, we set the different number to r according to the level number in a sequential order. To make the r array satisfy the circuits of different sizes as many as possible, we made statistic doing the experiments to test benches in [15] and [16]. Finally, array $\{0, 3, 2, 2, 2, 1, 1\}$ was decided. Fig. 7 gives an example, the grids with black point are effective grids. At first, only an initial effective grid exists, so we extend the detective region by Algorithm 1 radius by radius. The extension time is 3, 2, 2, 2, 1, 1 as the coarsening level increases. More and more effective grids are contained and the capacity becomes sufficient.

IV. Thermal Model and Thermal TSV

The thermal model makes use of the heat dissipation path shown in Fig. 8. Heat could dissipate along the vertical direction. We introduce a significant concept named *thermal*

Algorithm 1 Conditional Grid Extension

Require: $grid(x_0, y_0)$
Ensure: the set of extended grids $V = (x, y)$
$d_{max}\_b[] = \{0, 3, 2, 2, 2, 1, 1, 0, 0\}$
$d_{max} \leftarrow d_{max}\_b[level]$
$d \Leftarrow 0$
for $d \leqslant d_{max}$ **do**
  $i \Leftarrow 0$
  for $i \leqslant d$ **do**
    $V \Leftarrow V + (x_0 - i, y_0 + d + 1)$
    $V \Leftarrow V + (x_0 + i + 1, y_0 + d - i + 1)$
    $V \Leftarrow V + (x_0 - i, y_0 - d + i)$
    $V \Leftarrow V + (x_0 + i + 1, y_0 - d + i)$
    $i \Leftarrow i + 1$
  end for
  $d \Leftarrow d + 1$
end for
return V



(a)

(b)

Fig. 9. Detailed resistive network and flow conservation law. (a) Resistive network of one chip die with 4×4 grid size. (b) Incoming current for node i.

resistance and could calculate by

$$R = \frac{T_2 - T_1}{P} \tag{3}$$

where T_1 and T_2 are the temperature at two edges of a heated object, P is the power of thermal source. $g = (1/R)$ is called *thermal conductance*.

A. Thermal Resistive Model

Resistive thermal model compare the thermal condition to electric model, that is to say, compare the temperature to the voltage, the power to the electric charge, the thermal resistance to the resistance, the thermal conductance to the conductance, the constant temperature source to the constant voltage source. We use conservation of electric charge theorem to help calculate the circuit. For wire length and temperature results from work [1], the assignment difference in different algorithms can be more easily identified.

Fig. 9(a) shows an example of a resistive network for a chip with 4×4 grid size. Every node is represented as vertex i and thermal resistive $g_{i,j}$ connects the adjacent vertices. v_i represents the temperature of vertex i. The heat dissipation conductivity g_i and voltage source $\overline{V_i}$ are related to room temperature. Current source I_i represents the heat generated by inserted signal TSV in each grid, which is proportional to number of TSV in the grid. The denotation of symbols is as follows.

1) i: Grid i.
2) $\overline{V_i}$: Constant room temperature.
3) g_i: Constant heat dissipation conductivity of node i.
4) v_i: Temperature of node i that needs evaluating.
5) I_i: Heat generated at node i that changes according to the signal TSV number in the corresponding grid.
6) $g_{i,j}$: Thermal conductivity between node i and j that is constant between the adjacent grids and 0 otherwise.

The initial placement of nets is already determined by former steps (multilevel assignment). Then, we use *Kirchhoff's circuit law* to build equation at nodes to calculate the temperature. Following the *current conservation law*, current flows into a node should be equivalent to that flows out. As Fig. 9(b) depicts, in a chip consists of $m \times n$ grids, for node i there are currents flow from voltage source, current source, and six adjacent nodes, i.e., $i-1$, $i+1$, $i-n$, $i+n$, $i-mn$, and $i+mn$ from left, right, front, behind, top, and bottom direction, respectively. The equation could be written as

$$\begin{aligned}&(v_{i-1}-v_i)g_{i-1,i}+(v_{i-n}-v_i)g_{i-n,i}+(v_{i+1}-v_i)g_{i,i+1}\\&\quad+(v_{i+n}-v_i)g_{i,i+n}+(v_{i-mn}-v_i)g_{i-mn,i}\\&\quad+(v_{i+mn}-v_i)g_{i,i+mn}+(\overline{V_i}-v_i)g_i+I_i=0.\end{aligned}\tag{4}$$

We get such equation for each chip die of $m \times n$ grids. What's more, we put the equations of all of vertices in a matrix equation like (5). Here, G is an $N \times N$ thermal conductivity matrix, where $N = m \times n$. In the matrix G, $g_{ij} = g_{i-1,i} + g_{i-n,i} + g_{i,i+1} + g_{i,i+n} + g_{i-mn,i} + g_{i,i+mn} + g_i$ when $i = j$; while $i \neq j$, if node i and j are adjacent, $g_{ij} = -g_{ij}$, otherwise $g_{ij} = 0$. We get matrix equation of one chip die as follows:

$$\begin{bmatrix} g_{11} & g_{12} & \cdots & g_{1N} \\ g_{21} & g_{22} & \cdots & g_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ g_{N1} & g_{N2} & \cdots & g_{NN} \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{bmatrix} = \begin{bmatrix} I_1+\overline{V}\cdot g_1 \\ I_2+\overline{V}\cdot g_2 \\ \vdots \\ I_N+\overline{V}\cdot g_N \end{bmatrix}. \tag{5}$$

Due to the multidie in the chip, denoting the number of dies as l, we combine the equation of each die to one matrix equation. In this way, we solve the matrix equation of each chip die so that we get temperature v_i of each grid

$$G \cdot V = I + \overline{V} \cdot g = \overline{I}.$$

B. Effect of Thermal TSV

Here, a structure called *thermal TSV* is introduced. Considering the heat dissipation path, the thermal TSV which passes across the silicon layer in the vertical direction is effective in dissipating the heat generated by components. The number of thermal TSVs decides the effect of temperature reduction. Provided there are ttsv_{num} thermal TSVs on the grid of 3-D-IC and the thermal conductance of each thermal TSV is g_{TSV}.

Algorithm 2 shows the method of optimizing the overheated regions. On the last step, we get the temperature distribution on each die. In order to reduce the temperature of some overheated grids whose v_i is larger than threshold v_{th}, we insert thermal TSVs to those regions. v_{th} is decided by the value of 10% highest temperature of grids on the top layer. Counting the number of different types of components in the grid, we sum up the total area of components meanwhile get the available area to insert thermal TSV. Due to the *constraint of available area*, in some occasions, we cannot insert enough TSVs to make v_i not be higher than v_{th} so that we determine the maximum value of ttsv_{num} in the following equation:

$$\text{ttsv}_{\text{num}} = \frac{\text{grid area} - \text{total area of components}}{\text{thermal TSV area}}. \tag{6}$$

Algorithm 2 Thermal Optimization Method

Require: Circuit properties and statistics
Ensure: Final temperature distribution
1: Establish thermal resistive network
2: Present the matrix equation
3: Get the vector $V = v_1, v_2, \cdots, v_{l \cdot N}$
4: Regard v_{th} as the temperature of 10% highest temperature on top layer
5: **while** each grid **do**
6: **if** $v_i \geq v_{th}$ **then**
7: Regard g_i as overheated grid
8: Calculate the total area of components in g_i
9: Determine the maximum $ttsv_{num}$ in g_i
10: **end if**
11: **end while**
12: **while** each overheated grid **do**
13: Regard $g_{ij} + g_{TSV} * ttsv_{num}$ as new g_{ij} and substitute in the matrix equation
14: **end while**
15: Get the vector $V = \{v_1, v_2, \cdots, v_{l \cdot N}\}$
16: **return** V

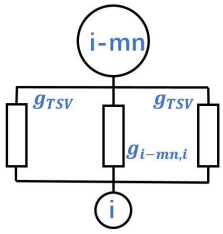


Fig. 10. Effect of inserted thermal TSV in the circuit.

In the matrix equation, we substitute g_{ij} as $g_{ij} + g_{\text{TSV}} * \text{ttsv}_{\text{num}}$ for all of the grids that are inserted thermal TSVs. Because the inserted thermal TSV in the circuit brings *parallel thermal conductance* as Fig. 10 shows. Finally, solve the equation and we get the vector V after optimization as optimized temperature distribution.

V. Multipins Assignment

A. Overview of Multipins Assignment

In the assignment problem, we usually denote each net in the netlist as $n = (s, t)$ which has only two pins. The assumption simplifies the procedure and calculations making the model simple, while it greatly obeys the reality. The real circuit consists of many pins in a net and in our problem, we denote the net as $n = (s^1, s^2, s^3, \ldots, t)$. Here, s is the pin on a certain layer of chip that could be multiple and t is the pin where a net connects to the bottom layer, which a net should have only one.

In order to realize the connectivity of the pins, we make use of *RST*. Different from MST, RST allows to add extra vertexes when connecting vertexes making the weighting sum of the network minimum. MST could be regarded as a special occasion of RST. First, we project the vertices of the net onto one plane. Then, we could construct a 2-D RST of the net like Fig. 11(a) by inserting extra vertices as TSVs.

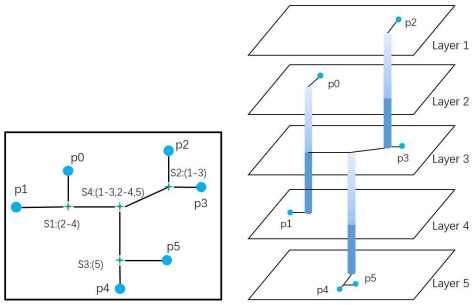


Fig. 11. Construction of a 3-D RST. (a) 2-D RST. (b) 3-D RST.

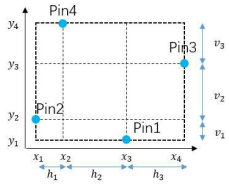


Fig. 12. Nets are shown in Hanan grid.

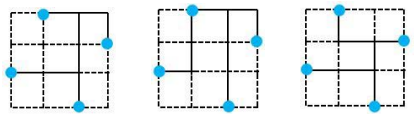


Fig. 13. Three instances of RST in Hanan grid. (a) (1,2,1,1,1,2). (b) (1,1,1,1,2,3). (c) (1,2,1,1,1,1).

After accomplishing the connectivity, we transfer 2-D RST back to 3-D graph and get the 3d RST result shown in Fig. 11(b).

B. Establishing Rectilinear Steiner Tree

We introduce the *FLUTE* method in [11] to help establish 2-D RST. Note that the length of a horizontal (vertical) edge in the *Hanan grid* is the distance between two adjacent horizontal (vertical) Hanan lines, i.e., $h_i = x_{i+1} - x_i$, $v_i = y_{i+1} - y_i$. The definition is shown in Fig. 12. The RST is decomposed to represent a bunch of Hanan edges and we could transfer the RST graph in linear combination using Hanan. For example, in Fig. 13(a)–(c), the net could be written as $h_1+2h_2+h_3+v_1+v_2+2v_3$, $h_1+h_2+h_3+v_1+2v_2+3v_3$, $h_1+2h_2+h_3+v_1+v_2+v_3$. Furthermore, we transfer the combination to wirelength vector, (1, 2, 1, 1, 1, 2), (1, 1, 1, 1, 2, 3), (1, 2, 1, 1, 1, 1). It becomes more specific to compare and calculate by labeling in vector. We could easily judge the wirelength in Fig. 13(a) is more than Fig. 13(c) because the value of v_3 is larger. In this way, it is viable to find the minimum RST from many kinds of possibility by easily calculating.

C. Die Span Computation

When we think about the *die span*, there are three occasions in different position relation between two TSVs illustrated in Fig. 14. We set temporary variable $tTop$ to record the minimum location of TSV *bot* and $tBot$ to record maximum location of TSV *top*. $tTop$ and $tBot$ make up the die span. For example, in Fig. 14(a), the $tTop$ is smaller than $tBot$, we use a

TABLE I
FLOOR PLAN DATA

Test Bench	Grid Size	Layer	#Net	#g_exist	#max_cap
ami33	35×35	3	85	750	5
	48×48	3	85	1418	5
	64×64	4	85	3360	5
ami49	26×26	3	207	416	6
	38×38	3	207	900	6
	52×52	3	207	1657	5
	60×60	4	207	2970	7
n50	33×33	3	382	676	8
	41×41	4	382	1383	8
	60×60	5	382	3722	8
n100	38×38	4	676	1184	8
	49×49	4	676	1975	8
	56×56	4	676	2591	8
n200	28×28	3	848	475	10
	42×42	4	848	1439	8
n300	41×41	4	1246	1355	12
	46×46	4	1246	1739	10
	56×56	5	1246	3224	10
YMC	32×32	4	1500	828	4

TABLE II
THERMAL CONDITION OF YMC

Layer	Min. T	Max. T	Original Ave. T	Optimized Ave. T	v_{th}	#Thermal TSV
1	8.8	85.2	43.5	42.7	59.9	137
2	8.3	76.5	41.9	41.3	57.3	102
3	7.9	78.4	41.6	41.2	54.6	68
4	7.8	67.0	39.8	39.6	49.6	34

TABLE III
COMPARISON OF MST-BASED AND RST-BASED FLOORPLAN

Test Bench	#dies	3D MST-based		3D RST-based	
		HPWL($\times 10^5 \mu m$)	#TSVs	HPWL($\times 10^5 \mu m$)	#TSVs
n100	3	0.168(1.00)	1349(1.00)	0.156(0.93)	1165(0.86)
	4	0.226(1.00)	2215(1.00)	0.208(0.92)	1841(0.83)
	5	0.294(1.00)	2779(1.00)	0.258(0.88)	2258(0.81)
AVE		(1.00)	(1.00)	(0.91)	(0.83)
n200	3	0.621(1.00)	3906(1.00)	0.539(0.87)	3274(0.84)
	4	0.692(1.00)	5800(1.00)	0.609(0.88)	4771(0.82)
	5	0.855(1.00)	7538(1.00)	0.757(0.89)	5981(0.79)
AVE		(1.00)	(1.00)	(0.88)	(0.82)
n300	3	0.993(1.00)	4876(1.00)	0.886(0.89)	4111(0.84)
	4	1.234(1.00)	7538(1.00)	1.096(0.88)	6202(0.82)
	5	1.172(1.00)	9860(1.00)	10528	1.027(0.88)
AVE		(1.00)	(1.00)	(0.88)	(0.82)

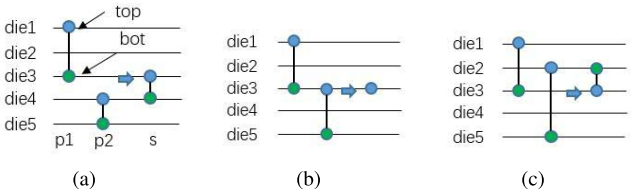


Fig. 14. Die span diagram. (a) $tTop < tBot$. (b) $tTop = tBot$. (c) $tTop > tBot$.

TSV to connect $tTop$th layer with $tBot$th; in Fig. 14(b), $tTop$ equals to $tBot$, we could directly use planar wires to connect two TSVs; in Fig. 14(c), $tTop$ is larger than $tBot$, there is overlap of layers so that we need not add TSVs between layers, but use planar wires to connect two TSVs on whichever [$tTop$, $tBot$] layer.

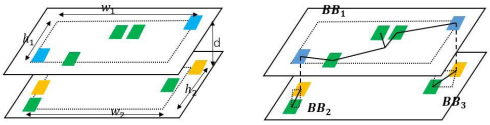
D. Calculation of Wirelength

One of the way to calculate the wirelength is to construct a 2-D bounding box which includes width w and height h of chip, and distance between two dies d. We sum the *half perimeter wirelength (HPWL)* of each die and vertical distance of TSVs up. However, 2-D bounding box is not accurate enough, [18] uses another wirelength metric HPWL-3-D. Each subnet has its own bounding box, we calculate the wirelength

TABLE IV
EVALUATION OF CONDITIONAL GRID EXTENSION ALGORITHM

Test Bench	Grid Size	#Nets	Without Using		Unconditional		Conditional		
			Wirelength	Overflow[1]	Wirelength	Time (ms)	Wirelength	Overflow	Time (ms)
ami33	35×35	85	2469(1.00)	28(1.00)	2142 (0.87)	124(1.00)	1936 (0.78)	1 (0.04)	78 (0.63)
ami49	26×26	207	4306(1.00)	23(1.00)	4385 (1.02)	203(1.00)	4356 (1.01)	8 (0.38)	179 (0.88)
	38×38	207	5522(1.00)	111(1.00)	5420 (0.98)	378(1.00)	5236 (0.95)	3 (0.027)	221 (0.59)
n50	33×33	382	9178(1.00)	60(1.00)	8955 (0.98)	755(1.00)	8745 (0.95)	4 (0.07)	546 (0.72)
	41×41	382	11569(1.00)	3(1.00)	10726 (0.93)	1149(1.00)	10140 (0.88)	1 (0.33)	719 (0.63)
n200	28×28	848	14171(1.00)	293(1.00)	13453 (0.95)	1730(1.00)	13202 (0.93)	176 (0.60)	1284 (0.74)
n300	41×41	1246	36431(1.00)	63(1.00)	33778 (0.93)	7874(1.00)	33123 (0.91)	13 (0.21)	5405 (0.69)
YMC	32×32	1500	17547(1.00)	863(1.00)	16652 (0.95)	1944	16312 (0.93)	769 (0.89)	1447 (0.74)
AVE			1.00**X**	1.00**X**	0.95**X**	1.00**X**	0.92**X**	0.67**X**	0.70**X**

[1] Overflow is the excessive flow that cannot be allocated to effective grids in un-coarsening assignment.



(a) (b)

Fig. 15. Wirelength metrics for 3-D nets. (a) HPWL−2DBB $= \sum(h_i + w_i) + 2d$. (b) HPWL − 3D $= 2d + \sum(BB_i)$.

of a 3-D net in the following equation:

$$\text{HPWL} - \text{3-D}(n_i) = d * N_{\text{TSV},i} + \sum \text{HPWL}(BB_{i,j}) \quad (7)$$

where d is the distance of a TSV, $N_{\text{TSV},i}$ is the total number of TSVs in net n_i, HPWL$(BB_{i,j})$ is the HPWL of jth bounding box of n_i.

VI. EXPERIMENT RESULTS

A. Experiment Introduction

1) Experimental Environment: We test our algorithm using C programing language on the 2.5-GHz Intel(R) Core(M) workstation with 8-GB memory. The experiments are carried on two Microelectronics Center of North Carolina [16] and four Giant Scale Resource Circuit [17] benchmarks, including ami33, ami49, n50, n100, n200, n300, Yoshimura circuit (YMC), and the floor plans are generated randomly. In all of test benches we have tried different grid sizes to do the experiments. The data of each test bench are given in Table I. The meaning in the table is as follows.

1) *Grid size:* Each layer is divided into $g_1 \times g_2$ grids.
2) *Layer:* The number of chip dies of 3-D-IC.
3) *#net:* The total number of nets in the netlists.
4) *#g_exist:* The number of grids with capacity, which could contain TSVs.
5) *#max_cap:* The maximum capacity of grids.

B. Comparison and Evaluation

Table II shows the average temperature and number of inserted TSV in each layer. It proves the effectiveness of temperature reduction by applying thermal resistive model and inserting thermal TSV. The reduction effect is related to the property of circuits. Generally, the higher the ratio of a number of components to board area is, the more is the temperature reduction. We could get the conclusion that the higher the layer location is, the higher is the temperature.

Fig. 16 depicts the temperature distribution on each layer of YMC on a 2-D isothermal level graph. The temperature

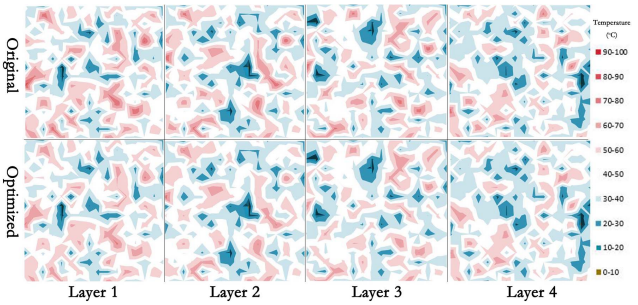


Fig. 16. Comparison of temperature distribution on each layer of YMC before/after inserting thermal TSVs.

is increasing as the color going to red from blue. According to the heat dissipation path and temperature increase formula, we insert a number of thermal TSV in the proper region. It is obvious that the color of some regions of the circuits fades proving the heat dissipation path makes sense. The maximum temperature decreases as thermal TSVs are inserted in the overheated regions. The median temperature decreases as well, which results from thermal TSV improves the temperature condition of all the regions it passes through.

Table III shows the results when floorplaning using MST and RST, respectively. By using RST, the HPWL-3-D could be reduced 10% around and the number of TSVs is 18% less. The improvement is caused by RST's routing algorithm. Different from MST, RST could insert extra vertices into the graph and find the optimal TSV locations using less TSVs and shorter wirelength. We could also get the conclusion that with the number of dies goes up, the improvement by RST becomes more obvious.

In Table IV, the conditional grid extension and unconditional grid extension together with the previous work [5] are evaluated. By using the grid extension method, we could assign more TSVs on effective grids, which increases efficiency and avoids unnecessary residual path calculation. It shows on average 32.7% overflows are eliminated. Wirelength is also reduced because we can easily get shorter path led by capacity limits. As the chart shows, unconditional grid extension reduces the wirelength by 5.1% and conditional grid extension reduces by 8.2% compared with [5]. Although in some case, like ami49 26 × 26, the result becomes worse. There is a trend that the circuits with lager grid size tend to get shorter wirelength. Moreover, conditional grid extension is better than the unconditional one, because grid extension

is conducted in the iterative way which is so time consuming and conditional method saves the resources in the high-level coarsened graph. Conditional grid extension method could save the time cost by 30.2% on average.

VII. Conclusion

The experiments show that methods I propose could provide a better solution in many field, such as wirelength, time consuming, and temperature.

1) The unconditional grid extension and conditional grid extension could reduce the wirelength by 5.1% and 8.2%, respectively. Compared with and without using extension method, conditional grid extension could eliminate 32.7% overflows and compared with unconditional grid extension could save time cost by 30.2%.
2) We could get explicit heat distribution of chips and thermal TSVs could effectively reduce the temperature.
3) In the assignment of multipins, using HPWL-3-D could reduce HPWL by 30% around in some large circuits and establishing RST makes HPWL 10% less and the number of TSVs 18% less.

References

[1] Y. Qian, C. Hao, and T. Yoshimura, "3D-IC signal TSV assignment for thermal and wirelength optimization," in *Proc. 27th Int. Symp. Power Timing Modeling, Optim. Simulation (PATMOS)*, Sep. 2017, pp. 1–8, doi: 10.1109/PATMOS.2017.8106948.

[2] J. Cong and Y. Zhang, "Thermal-driven multilevel routing for 3D ICs," in *Proc. Asia South Pacific Design Automat. Conf., (ASP-DAC)*, Jan. 2005, pp. 121–126.

[3] H.-H. Yeh, C.-Y. Huang, and S.-H. Huang, "Temperature rise minimization through simultaneous layer assignment and thermal through-silicon-via planning," in *Proc. 8th Int. Microsyst., Packag., Assem. Circuits Technol. Conf. (IMPACT)*, Oct. 2013, pp. 207–210, doi: 10.1109/IMPACT.2013.6706671.

[4] J. Cong, M. Xie, and Y. Zhang, "An enhanced multilevel routing system," in *Proc. IEEE/ACM Int. Conf. Comput. Aided Design (ICCAD)*, Nov. 2002, pp. 51–58, doi: 10.1109/ICCAD.2002.1167513.

[5] C. Hao, N. Ding, and T. Yoshimura, "An efficient algorithm for 3D-IC TSV assignment," in *Proc. 14th IEEE Int. New Circuits Syst. Conf. (NEWCAS)*, Jun. 2016, pp. 1–4, doi: 10.1109/NEWCAS.2016.7604740.

[6] C. Hao and T. Yoshimura, "An efficient multi-level algorithm for 3D-IC TSV assignment," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E100A, no. 3, pp. 776–784, Mar. 2017.

[7] D. H. Kim, K. Athikulwongse, and S. K. Lim, "Study of through-silicon-via impact on the 3-D stacked IC layout," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 21, no. 5, pp. 862–874, May 2013, doi: 10.1109/TVLSI.2012.2201760.

[8] X. He, S. Dong, and Y. Ma, "Signal through-the-silicon via planning and pin assignment for thermal and wire length optimization in 3D ICs," *Integr., VLSI J.*, vol. 43, no. 4, pp. 342–352, 2010.

[9] S. Chen, L. Ge, M.-F. Chiang, and T. Yoshimura, "Lagrangian relaxation based inter-layer signal via assignment for 3-D ICs," *IEICE Trans. Fundam. Electron., Commun. Comput. Sci.*, vol. E92.A, no. 4, pp. 1080–1087, Apr. 2009.

[10] D. H. Kim, R. O. Topaloglu, and S. K. Lim, "Block-level 3D IC design with through-silicon-via planning," in *Proc. 17th Asia South Pacific Design Automat. Conf.*, Jan./Feb. 2012, pp. 335–340, doi: 10.1109/ASP-DAC.2012.6164969.

[11] C. Chu and Y.-C. Wong, "FLUTE: Fast lookup table based rectilinear Steiner minimal tree algorithm for VLSI design," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 27, no. 1, pp. 70–83, Jan. 2008.

[12] M. Koyanagi, T. Fukushima, and T. Tanaka, "High-density through silicon vias for 3-D LSIs," *Proc. IEEE*, vol. 97, no. 1, pp. 49–59, Jan. 2009, doi: 10.1109/JPROC.2008.2007463.

[13] M. S. Bakir *et al.*, "3D heterogeneous integrated systems: Liquid cooling, power delivery, and implementation," in *Proc. IEEE Custom Integr. Circuits Conf. (CICC)*, Sep. 2008, pp. 663–670, doi: 10.1109/CICC.2008.4672173.

[14] X. Liu, G. Yeap, J. Tao, and X. Zeng, "Integrated algorithm for 3-D IC through-silicon via assignment," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 22, no. 3, pp. 655–666, Mar. 2014, doi: 10.1109/TVLSI.2013.2246876.

[15] H.-H. Yeh, S.-H. Huang, and K.-H. Li, "3D IC design partitioning for temperature rise minimization," in *Proc. 6th Int. Microsyst., Packag., Assem. Circuits Technol. Conf.*, Oct. 2011, pp. 447–450, doi: 10.1109/IMPACT.2011.6117253.

[16] CBL. *The Physical Design 1991 Benchmark (MCNC Benchmarks)*. Accessed: Oct. 22, 2007. [Online]. Available: http://www.cbl.ncsu.edu:16080/benchmarks/PDWorkshop91/bench/

[17] GSRC. *Floorplan Benchmarks*. [Online]. Available: http://vlsicad.eecs.umich.edu/BK/BlockPacking/

[18] X. He, S. Dong, Y. Ma, and X. Hong, "Simultaneous buffer and interlayer via planning for 3D floorplanning," in *Proc. 10th Int. Symp. Qual. Electron. Design*, Mar. 2009, pp. 740–745, doi: 10.1109/ISQED.2009.4810385.

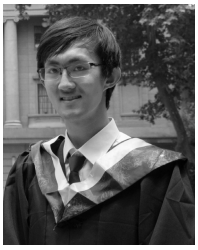
[19] M.-C. Tsai, T.-C. Wang, and T. Hwang, "Through-silicon via planning in 3-D floorplanning," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 8, pp. 1448–1457, Aug. 2010.

[20] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. Cambridge, MA, USA: MIT Press, 2009.

[21] J. P. Gambino, S. A. Adderly, and J. U. Knickerbocker, "An overview of through-silicon-via technology and manufacturing challenges," *Microelectron. Eng.*, vol. 135, pp. 73–106, Mar. 2015.

[22] K.-J. Wolter, "System integration by advanced electronics packaging," in *Bio and Nano Packaging Techniques for Electron Devices*, G. Gerlach and K.-J. Wolter, Eds. Berlin, Germany: Springer-Verlag, 2012, ch. 12.

[23] M. Dellutri *et al.*, "1 Gb stacked solution of multilevel NOR flash memory packaged in a LFBGA 8 mm by 10 mm by 1.4 mm of thickness," in *Proc. 7th Int. Conf. Therm., Mech. Multiphys. Simulation Exp. Micro-Electron. Micro-Syst. (EuroSime)*, Apr. 2006, pp. 1–5, doi: 10.1109/ESIME.2006.1643997.

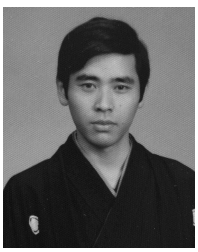


Yi Zhao received the B.S. degree in electronic science and engineering from Southeast University, Nanjing, China, in 2017. He is currently pursuing the M.S. degree with the Graduate School of IPS, Waseda University, Kitakyushu, Japan.

His current research interests include large-scale integrated design and optimization.

Cong Hao (M'16) received the B.S. degree in computer science and the M.S. degree from Shanghai Jiao Tong University, Shanghai, China, in 2011 and 2014, respectively, and the M.S. degree from the Graduate School of IPS, Waseda University, Kitakyushu, Japan, in 2012, She is currently pursuing the Ph.D. degree with the Graduate School of IPS, Waseda University, Kitakyushu.

Her current research interests include very large scale integration design automation, Networkon-Chip, and reconfigurable architecture.



Takeshi Yoshimura (M'86) received the B.E., M.E., and Dr.Eng. degrees from Osaka University, Suita, Japan, in 1972, 1974, and 1997, respectively.

From 1979 to 1980 , he was on leave with the Electronics Research Laboratory, University of California, Berkeley, CA, USA, where he was involved in very large scale integration CAD. In 2003, he joined the Graduate School of Information, Production and Systems, Waseda University, Kitakyushu, Japan.