# Accurate TSV Minimization in High-Level Synthesis of 3D ICs Design

Chih-Hung Lee, Chun-Hua Cheng, and Shih-Hsu Huang
Department of Electronic Engineering,
Chung Yuan Christian University,
Chung Li, Taiwan, R.O.C.
Email: shhuang@cycu.edu.tw

## ABSTRACT

Recent progress in manufacturing technology makes it is possible to vertically stack multiple integrated chips, to develop CAD tools according to characteristics of 3D architecture is urgent and important. In this paper, we propose an integer linear programming formulation to perform signal through-the-silicon vias (TSV) minimization in high-level synthesis of 3D ICs. Different from previous technical literatures [1] [2] that vias number is minimized with a complementary objective; our formulation directly minimizes the accurate vias number. Since vias number is determined by layer assignment result of communicating resources rather than communicating operations, experimental results promise that our formulation is more effective and accurate on via minimization than previous technical literatures.

## 1. INTRODUCTION

Three dimensional integrated circuits (3D ICs) technologies [3] [4] can highly integrate systems by vertically stacking and connecting various materials, technologies and functional units together. The benefits of 3D ICs include high performance, low power, cheap package, flexible heterogeneous integration in comparison with 2D ICs. For long-line interconnections in an original 2D ICs design, the signal transmission performance cloud be enhanced by short vertical interconnects (through-the-silicon via, TSV) under the 3D-IC structure [5] . And TSV can also decrease the maximum temperature over 50% [6] . However, the area size of a TSV is much larger than a regular via [7] and TSVs not only consume silicon area but the additional area for TSVs may increases total chip area. Moreover, TSVs act as obstacles during placement and routing [8] . Since TSVs are expensive to fabricate, the minimization on total number of TSVs in a circuit is an important issue.

In physical design stage, researches on vias planning/minimization are provided with various viewpoints. Via minimization with temperature constraints problem is formulated as a constrained nonlinear problem based on the thermal resistive model and solved in a multilevel framework [9] . In [10] , the thermal conductivity and thermal via density of each silicon are optimized after placement. The first thermal-driven 3-D multilevel routing algorithm which features an integrated adaptive lumped resistive thermal model and a simultaneous multilevel TS-via planning [11] , and experimental results show their approaches are effective in reducing the dummy TSV to bring down the circuit temperature to a required level with similar wirelength and with a reasonable increase in runtime.

For high-level synthesis, Mukherjee et al. propose an integer linear programming formulation to perform simultaneous and optimal scheduling, binding and layer assignment for synthesizing designs [1] . Various of cost functions aimed at

minimizing the number of vias/length of interconnect in the critical path are provided in [2] . Krishnan et al. proposed a SA-based layout aware binding algorithm by integrating a binder and a floorplanner [12] , the objective cost is a linear combination of footprint area, the total wirelength, the difference in floorplan dimensions among individual dies in the 3-D IC, and the number of die-to-die vias. However, their approach separately generates results of resource binding and layer assignment with two representations. Up to now, ILP-based researches on high level synthesis of 3-D ICs, Via minimization is formulated with a complementary objective that tries to maximize the assignment of all pairs of communicating operations to the same layer [1] [2] . In fact, since via is directly decided by the communicating resources rather than communicating operations, the existed approaches that try to minimize via number by maximizing the assignment of all pairs of communicating operations to the same layer can not guarantee the optimality on via minimization due to without considering the impact of layer assignment result of resources. Thus, the TSV minimization is inaccurate.

In this paper, we propose an integer linear programming formulation that directly to minimize total TSV numbers in high-level synthesis stage. Experimental results show that our approach can further reduce 30.26% total number of TSVs, without any overhead on the footprint area. The remainder of this paper is organized as follow. In Section 2, we provide the motivation to study real TSV minimization problem. In section 3, we propose an integer linear programming approach to formally draw up accurate TSV minimization in high-level synthesis of 3D ICs design. In section 4, we report the result of TSV minimization in high-level synthesis of 3D ICs. Finally, in section 5, we make a conclusion.

## 2. MOTIVATIONAL EXAMPLE

Let's use the scheduled DFG HAL shown in Figure 1 for illustration. Suppose that we are given one adder, called $A_1$, one subtraction, called $S_1$, two multipliers, called $M_1$ and $M_2$, and one comparator, called $C_1$. Here, we give three layers to assign those resources. The objective is to reduce total number of TSVs. We know that [1] and [2] maximize all pairs of communicating operations which are assigned to the same layer. Suppose that operations $o_3$ and $o_{11}$ share adder $A_1$, operations $o_6$ and $o_{10}$ share subtraction $S_1$, operation $o_1$, $o_5$, and $o_8$ are assigned to multiplier $M_1$, operation $o_2$, $o_4$, and $o_7$ are assigned to multiplier $M_2$, and operation $o_9$ is assigned to comparator $C_1$. We use the following form to describe this resource binding solution: $A_1 = \{o_3, o_{11}\}$, $S_1 = \{o_6, o_{10}\}$, $M_1 = \{o_1, o_5, o_8\}$, $M_2 = \{o_2, o_4, o_7\}$, and $C_1 = \{o_9\}$. We use [2] to find the layer assignment solution as shown in Figure 2(a) that the adder $A_1$ and $C_1$ are assigned to layer 1, the multiplier $M_1$ is assigned to layer 2, and the multiplier $M_2$ and subtraction $S_1$ are assigned to layer 3. We know that the objective [2] is to find the maximum pairs of communicating operations at the same layer. In this example, the pair of communicating operations $o_9 \rightarrow o_{11}$ is assigned to the layer 1, the pair of communicating operations $o_5 \rightarrow o_8$ is assigned to the layer 2, the pairs of communicating operations $o_2 \rightarrow o_4$, $o_4 \rightarrow o_7$, $o_7 \rightarrow o_{10}$ are assigned to the layer 3. The maximum pairs of communicating operations at the same layer are 5. From Figure 2(a), we know that the total number of TSVs is 3.

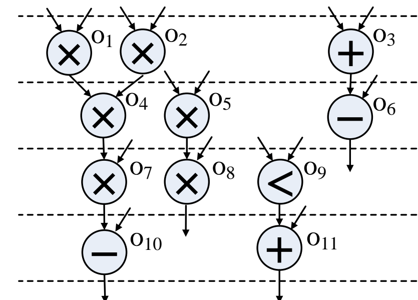


Figure 1: Scheduled DFG-HAL.

However, we have another solution that the maximum pairs of communicating operations are still 5, but the total number of TSVs is less than 3. We still use scheduled DFG shown in Figure 1 for example. Consider another resource binding solution: $A_1 = \{o_3, o_{11}\}$, $S_1 = \{o_6, o_{10}\}$, $M_1 = \{o_1, o_4, o_7\}$, $M_2 = \{o_2, o_5, o_8\}$, and $C_1 = \{o_9\}$. We have the resource layer assignment result shown as Figure 2(b) where the adder $A_1$ and $C_1$ are assigned to layer 1, the multiplier $M_1$ and subtraction $S_1$ are assigned to layer 2, and the multiplier $M_2$ is assigned to layer 3. In this example, we find that the pair of communicating operations $o_9 \rightarrow o_{11}$ is assigned to the layer 1, the pair of communicating operations $o_5 \rightarrow o_8$ is assigned to the layer 3, the pairs of communicating operations $o_1 \rightarrow o_4$, $o_4 \rightarrow o_7$ and $o_7 \rightarrow o_{10}$ are assigned to the layer 2. The maximum pairs of communicating operations at the same layer are still 5. However, from Figure 2(b), we find that the total number of TSVs is only 2.

Based on above observation, we know that total number of vias is determined by layer assignment result of communicating resources rather than communicating operations. Therefore, the objective function of [2] for TSV minimization is inaccurate. In this paper, we study accurate TSV minimization in high-level synthesis of 3D ICs design.

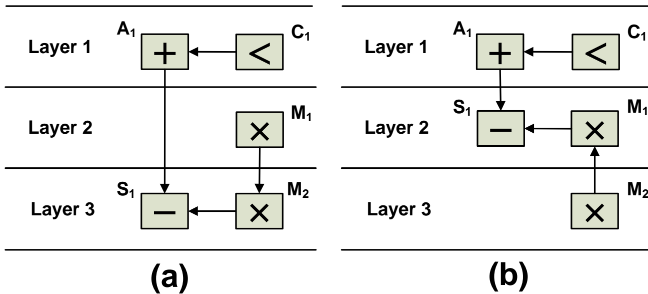


Figure 2: (a) Layer assignment I. (a) Layer assignment II.

## 3. ILP FORMULATION

In this section, we use an ILP to formally draw up our accurate TSV minimization in high-level synthesis of 3D ICs design. Given a DFG, constraints on the footprint area and the number of layers, our objective is to minimize total number of TSVs by performing the simultaneous application of resource allocation, scheduling, resource binding, and layer assignment. Note, under the given constraints, our approach guarantees obtaining the optimal solution.

Further, we define the following additional notations for our approach.

(1) The notation $D$ denotes the set that includes all the resources have communicating relation.

(2) The notation $via_{k1,k2}$ is a constant that denotes the number of TSVs between the resource $k_1$ and $k_2$.

(3) The notation $D_{k1,k2}$ is a binary variable. If the resource $k_1$ and $k_2$ have communicating relation, then $D_{k1,k2} = 1$; otherwise, $D_{k1,k2} = 0$.

(4) The value $C_{step}$ denotes the number of control steps in the data flow graph.

(5) The value $L$ denotes the number of active device layers in the design.

(6) The value $R_{max}$ denotes maximal number of resources in the design.

(7) $x_{i,j,k}$ is a binary variable that models if an operation $i$ is scheduled in control step $j$ and bind to resource $k$, then $x_{i,j,k} = 1$; otherwise, $x_{i,j,k} = 0$. The binary variable $x_{i,j}$ and $x_{i,k}$ can be derived from $x_{i,j,k}$. These variables model if an operation is scheduled in control step $j$, bind to resource $k$, respectively. The relations between them are: $x_{i,j} = \sum_{k=1}^{R\max} x_{i,j,k}$ and $x_{i,k} = \sum_{j=E_i}^{L_i} x_{i,j,k}$.

(8) The notation $r_{k,l}$ is a binary variable. If the resource $k$ is assigned to layer $l$, then $r_{k,l} = 1$; otherwise, $r_{k,l} = 0$.

(9) The value $A_k$ is the area of resource $k$ in the design.

(10) The value $A_{max}$ denotes the maximal area allowed for every layer.

(11) The value $p_k$ denotes the average power dissipation of resource $k$.

Next, we introduce our objective function and the constraints. Our objective function is to minimize total TSV numbers. Thus, we describe our objective function as below:

Minimize $\{\sum_{k_1 \in D} \sum_{k_2 \in D} via_{k_1,k_2}\}$.

If the operation $i_1$ and $i_2$ have communicating relation and the operation $i_1$ is assigned to resource $k_1$, operation $i_2$ is assigned to resource $k_2$, respectively, we know that the resource $k_1$ and $k_2$ have communicating constraint. Thus, for each operation $i_1$ and $i_2$, resource $k_1$ and $k_2$, we have the following constraint:

$$x_{i_1,k_1} + x_{i_2,k_2} \le 1 + D_{k_1,k_2}. \quad \text{(Formula 1)}$$

If the resource $k_1$ and $k_2$ have communicating relation (i.e., $D_{k1,k2} = 1$), then the number of TSVs determines on the equation $\sum_{l=1}^{L} l * r_{k_1,l} - \sum_{l=1}^{L} l * r_{k_2,l} \le via_{k_1,k_2}$. On the other hand, if the resource $k_1$ and $k_2$ do not have communicating relation (i.e., $D_{k1,k2} = 0$), then there is no TSVs constraint between resource $k_1$ and $k_2$. Let s denote a constant value that approximates infinity, i.e., $s \to \infty$, and thus does not restrict the number of TSVs. Thus, for each resource $k_1$ is assigned to layer $l_1$ and $k_2$ is assigned to layer $l_2$, we have the following constraint:

$$(\sum_{l_1=1}^{L} l_1 * r_{k_1,l_1} - \sum_{l_2=1}^{L} l_2 * r_{k_2,l_2}) - (1 - D_{k_1,k_2}) * s \le via_{k_1,k_2}, \quad \text{(Formula 2)}$$

$$(\sum_{l_2=1}^{L} l_2 * r_{k_2,l_2} - \sum_{l_1=1}^{L} l_1 * r_{k_1,l_1}) - (1 - D_{k_1,k_2}) * s \le via_{k_1,k_2}. \quad \text{(Formula 3)}$$

If the resource $k_1$ and $k_2$ have communicating relation, the number of their TSVs must be equal or bigger than 0. Thus, for each resource $k_1$ and $k_2$, we have the following constraint:

$$0 \le via_{k_1,k_2}. \quad \text{(Formula 4)}$$

A resource $k$ can be assigned to one and only one layer $l$. Thus, for each resource $k$, we have the following constraint:

$$\sum_{l=1}^{L} r_{k,l} = 1. \quad \text{(Formula 5)}$$

The sum of the areas of resource $k$ assigned to a layer $l$ must be less than or equal to the maximum area for a layer. Thus, for each layer $l$, we have the following constraint:

$$\sum_{k=1}^{R_{max}} A_k * r_{k,l} \le A_{max}. \quad \text{(Formula 6)}$$

Due to lifetime constraint, all operations can not share the same resource at the same control step. Thus, for each control step $j$ and resource $k$, we have the following constraint:

$$\sum_{i \in k} x_{i,j,k} \le 1. \quad \text{(Formula 7)}$$

An operation $i$ can be assigned to one and only one resource $k$. Thus, for each operation $i$, we have the following constraint:

$$\sum_{k=1}^{R\max} x_{i,k} = 1. \quad \text{(Formula 8)}$$

For each communicating relation operation $i_1$ and $i_2$, we have the following constraint:

$$\sum_{j=E_{i1}}^{L_{i1}} j * x_{i_1,j} < \sum_{j=E_{i2}}^{L_{i2}} j * x_{i_2,j}. \quad \text{(Formula 9)}$$

A decreasing power gradient is maintained from the lowest to the highest layers in order to control the thermal gradient in the design. Non-increasing power gradient between layers $l_1$ and $l_2$ where ($l_2 > l_1$) is enforced by the condition that for every pair of adjacent layers, the average power of resources in the upper layer must be less than that of the lower layer. Thus, for each layer $l_1$ and $l_2$, we have the following constraint:

$$\sum_{k=1}^{R\max} p_k * r_{k,l_1} \ge \sum_{k=1}^{R\max} p_k * r_{k,l_2}. \quad \text{(Formula 10)}$$

In the following, we use the DFG-HAL to illustrate our approach. Suppose that the resource constraints are one adder, one subtractor, two multipliers, and one comparator. The timing constraints are four control steps and layer assignment constraints are three layers. Our objective function is to minimize $\{via_{A1,S1} + via_{A1,M1} + via_{A1,M2} + via_{A1,M1} + via_{A1,C1} + via_{S1,A1} + via_{S1,M1} + via_{S1,M2} + via_{S1,C1} + via_{M1,A1} + via_{M1,S1} + via_{M1,M2} + via_{M1,C1} + via_{M2,A1} + via_{M2,S1} + via_{M2,M1} + via_{M2,C1}\}$.

We list all the constraints as below.

***Formula 1.*** Since the operation *2* and *4* have communicating relation, we have following constraints for all the control step and resource binding: $x_{o2,1,M1} + x_{o4,2,M2} \leq 1 + D_{M1M2}$; Similarly, we have the constraints $x_{o2,1,M2} + x_{o4,2,M1} \leq 1 + D_{M2M1}$; and so on.

***Formula 2.*** $(r_{M1,1}+2*r_{M1,2}+3*r_{M1,3}) - (r_{M2,1}+2*r_{M2,2}+3*r_{M2,3}) - (1 - D_{M1M2})*(L-1) \leq via_{M1,M2}$; Similarly, we have the constraints $(r_{M1,1}+2*r_{M1,2}+3*r_{M1,3}) - (r_{A1,1}+2*r_{A1,2}+3*r_{A1,3}) - (1 - D_{M1A1})*(L-1) \leq via_{M1,A1}$; and so on.

***Formula 3.*** $(r_{M2,1}+2*r_{M2,2}+3*r_{M2,3}) - (r_{M1,1}+2*r_{M1,2}+3*r_{M1,3}) - (1 - D_{M1M2})*(L-1) \leq via_{M1,M2}$; Similarly, we have the constraints $(r_{A1,1}+2*r_{A1,2}+3*r_{A1,3}) - (r_{M1,1}+2*r_{M1,2}+3*r_{M1,3}) - (1 - D_{M1A1})*(L-1) \leq via_{M1,A1}$; and so on.

***Formula 4.*** $0 \leq via_{M1,M2}$; Similarly, we have the constraints $0 \leq via_{M1,A1}$; and so on.

***Formula 5.*** $r_{M1,1}+r_{M1,2}+r_{M1,3} = 1$; Similarly, we have the constraints $r_{M2,1}+r_{M2,2}+r_{M2,3} = 1$; and so on.

***Formula 6.*** $A_{A1}*r_{A1,1} + A_{S1}*r_{S1,1} + A_{M1}*r_{M1,1} + A_{M2}*r_{M2,1}+ A_{C1}*r_{C1,1} \leq A_{max}$; Similarly, we have the constraints $A_{A1}*r_{A1,2}+A_{S1}*r_{S1,2}+A_{M1}*r_{M1,2}+A_{M2}*r_{M2,2}+ A_{C1}*r_{C1,2} \leq A_{max}$; and so on.

***Formula 7.*** $x_{o1,1,M1}+x_{o2,1,M1}+x_{o6,1,M1}+x_{o8,1,M1} \leq 1$; Similarly, we have the constraints $x_{o3,2,M1}+x_{o6,2,M1}+x_{o7,2,M1}+x_{o8,2,M1} \leq 1$; and so on.

***Formula 8.*** $x_{o1,1,M1} + x_{o1,1,M2} = 1$; Similarly, we have the constraints $x_{o2,1,M1} + x_{o2,1,M2} = 1$; and so on.

***Formula 9.*** $x_{o1,1,M1} + x_{o1,1,M2} < 2*x_{o3,2,M1} + 2*x_{o3,2,M2}$; Similarly, we have the constraints $x_{o6,1,M1} + 2*x_{o6,2,M1} + x_{o6,1,M2} + 2*x_{o6,2,M2} < 2*x_{o7,2,M1} + 3*x_{o7,3,M1} + 2*x_{o7,2,M2} + 3*x_{o7,3,M2}$.

***Formula 10.*** $p_{A1}*r_{A1,1}+ p_{S1}*r_{S1,1}+ p_{M1}*r_{M1,1}+ p_{M2}*r_{M2,1}+ p_{C1}*r_{C1,1} \leq p_{A1}*r_{A1,2}+ p_{S1}*r_{S1,2}+ p_{M1}*r_{M1,2}+ p_{M2}*r_{M2,2}+ p_{C1}*r_{C1,2}$. Similarly, we have the constraints $p_{A1}*r_{A1,1}+ p_{S1}*r_{S1,2}+ p_{M1}*r_{M1,2}+ p_{M2}*r_{M2,2}+ p_{C1}*r_{C1,2} \leq p_{A1}*r_{A1,3}+ p_{S1}*r_{S1,3}+ p_{M1}*r_{M1,3}+ p_{M2}*r_{M2,3}+ p_{C1}*r_{C1,3}$.

After solving these ILP formulations, we have that $x_{o1,1,M2} = x_{o2,1,M1} = x_{o3,1,A1} = x_{o4,2,M2} = x_{o5,2,M1} = x_{o6,3,S1} = x_{o7,3,M2} = x_{o8,3,M1} = x_{o9,3,C1} = x_{o10,4,S1} = x_{o11,4,A1} = x_{o2,1,M1} = r_{A1,1} = r_{C1,1} = r_{M1,2} = r_{M2,3} = r_{S1,3} = D_{A1C1} = D_{M1A1} = D_{M1M2} = D_{M2S1} = 1$, and the values of other binary variables are 0. The values of TSV variables $via_{M1,A1} = 1$ and $via_{M1,M2} = 1$, and the values of other TSV variables are 0. The total TSV numbers obtained by our ILP approach is only 2, which is also the lower bound of the total TSV numbers.

## 4. EXPERIMENTAL RESULTS

We use Extended LINGO Release 11.0 as the ILP solver. The platform is Windows 2003 x64 running on Intel Xeon E5355 CPU with 8GB RAM. Seven benchmark circuits are used to test the effectiveness of our approach. The benchmark TGFF14 is adopted from [13] . Benchmark circuits HAL [14] , BF [15] , AR [16] , G2 [17] and G5 [18] are popular DSP applications, while benchmark circuit R1 is control-dominated application adopted from [19] .

Without loss of generality, in our experiments, we assume that all the functional units and all the variables (registers) are 8-bit designs. The modules in the Synopsys DesignWare are adopted to implement the following functional units: adder, subtractor, multiplier, divisor, and comparator. Moreover, these functional units are targeted to TSMC 0.18μm process technology. The power consumption/area of adder, substrator, multiplier, divisor, and comparator are 428 μW / 4892 $\mu m^2$, 557 μW / 6326 $\mu m^2$, 1872 μW / 21455 $\mu m^2$, and 528 μW / 9147 $\mu m^2$, respectively.

Table 1 tabulates the characteristics of benchmark circuits. The column Operations gives 6-tuple (#ad,#su,#mu#,#di,#co), where #ad, #su, #mu, #di, and #co are the numbers of addition operations, subtraction operations, multiplication operations, and comparison operations, respectively. For example, benchmark circuit HAL has 2 addition operations, 2 subtraction operations, 6

multiplication operations, and 1 comparison operations. The column Steps gives the number of control steps. The column Functional Units gives 6-tuple (#add,#sub,#mul#,#div,#com), where #add, #sub, #mul#, #div, and #com are the numbers of adders, subtractors, multipliers, divisors, and comparators, respectively.

Table 2 tabulates the comparisons of the number of TSVs. We discuss them as below. The column Layer gives the number of layers. The column Area gives the footprint area. The column #via gives the number of TSVs. The column Time gives the CPU time in our approach. Since our objective function is accurate, in each benchmark circuit, our approach always has less total number of TSVs than [2]. In average, we find that our approach reduce 30.26% total number of TSVs.

## 5. CONCLUSIONS

This paper presents accurate TSV minimization in high-level synthesis of 3D ICs design. We use ILP to formally draw up the TSV minimization problem. Given constraints on the footprint area and the number of layers, our objective is to minimize total number of TSVs. Compared with previous work [2], benchmark circuits show that our approach can reduce 30.26% total number of TSVs, without any overhead on the footprint area.

| Circuit | Operations | Steps | Functional Units |
|---|---|---|---|
| TGFF14 | (7,3,3,1,0) | 7 | (2,1,1,1,0) |
| HAL | (2,2,6,0,1) | 4 | (1,1,2,0,1) |
| BF | (12,6,11,0,0) | 8 | (2,1,2,0,0) |
| AR | (12,0,16,0,0) | 10 | (2,0,3,0,0) |
| G2 | (9,0,9,0,6) | 10 | (2,0,2,0,1) |
| G5 | (24,0,0,0,4) | 10 | (4,0,0,0,1) |
| R1 | (55,0,15,0,12) | 11 | (6,0,7,0,5) |

**Table 1:** Characteristics of benchmark circuits.

| Design | **Design Constraints** | | **[2]** | | **Our** | |
|---|---|---|---|---|---|---|
| | Layer | Area ($\mu m^2$) | #via | Time (sec) | #via | Time (sec) |
| TGFF 14 | 2 | 27781 | 4 | <1 | 3 | <1 |
| | 3 | 26347 | 6 | <1 | 5 | <1 |
| | 4 | 21455 | 10 | <1 | 8 | <1 |
| HAL | 2 | 37965 | 1 | <1 | 1 | <1 |
| | 3 | 27781 | 3 | <1 | 2 | <1 |
| | 4 | 21455 | 7 | <1 | 5 | <1 |
| BF | 2 | 35412 | 3 | <1 | 3 | <1 |
| | 3 | 23608 | 13 | <1 | 8 | <1 |
| | 4 | 21455 | 18 | 1 | 11 | <1 |
| AR | 2 | 44490 | 5 | 5 | 3 | 2 |
| | 3 | 29660 | 5 | 56 | 3 | 10 |
| | 4 | 22245 | 20 | 29 | 10 | 10 |
| G2 | 2 | 37105 | 3 | 5 | 3 | 1 |
| | 3 | 24736 | 14 | < 1 | 8 | 17 |
| | 4 | 21455 | 15 | 1 | 10 | 56 |
| G5 | 2 | 17229 | 2 | 14 | 1 | 46 |
| | 3 | 11486 | 3 | 24 | 2 | 55 |
| | 4 | 9784 | 4 | 24 | 3 | 63 |
| R1 | 2 | 135163 | 1 | 323 | 1 | > 195 |
| | 3 | 90109 | 14 | 187 | 11 | 177 |
| | 4 | 67582 | 46 | 459 | 16 | 527 |

**Table 2.** Comparisons on the total number of TSVs

## 6. CONCLUSIONS

This paper presents accurate TSV minimization in high-level synthesis of 3D ICs design. We use ILP to formally draw up the TSV minimization problem. Given constraints on the footprint area and the number of layers, our objective is to minimize total number of TSVs. Compared with previous work [2], benchmark circuits show that our approach can reduce 30.26% total number of TSVs, without any overhead on the footprint area.

## 7. REFERENCES

[1] M. Mukherjee and R. Vemuri, "Simultaneous Scheduling, Binding and Layer Assignment for Synthesis of Vertically Integrated 3D Systems," Proc. of IEEE/ACM International Computer Design, pp. 222—227, 2004.

[2] M. Mukherjee and R. Vemuri, "On Physical-Aware Synthesis of Vertically Integrated 3D Systems," Proc. of International Conference on VLSI Design, pp. 647—647, 2005.

[3] S. Das, A. Fan, K.N. Chen, C.S. Tan, N. Checka and R. Reif, "Technology, Performance, and Computer-Aided Design of Three-dimensional Integrated Circuits," Proc. of international Symposium on Physical Design, pp. 108—115, 2004.

[4] S.S. Sapatnekar, "CAD for 3D Circuits: Solutions and Challenges," Proc. of the VLSI/ULSI Multilevel Interconnection Conference, pp. 245—251, 2007.

[5] R. Venkatesan, A. Kaloyeros, M. Beylansky, S.J. Souri, K. Banerjee, K.C. Saraswat, "Interconnect Limits on Gigascale Integration in the 21st Century," Proc. of the IEEE, vol 89, no. 3, pp. 305—324, 2001.

[6] S. B. Horn, "Vertically Integrated Sensor Arrays VISA (Invited)," Defense and Security Symposium, 2004.

[7] T.-Y. Chiang, S. J. Souri, C. O. Chui, and K. C. Saraswat, "Thermal Analysis of Heterogeneous 3-D ICs with Various Integration Scenarios," IEEE International Electron Devices Meeting, pp. 681—684, 2001.

[8] D.H. Kim, S. Mukhopadhyay, and S. K. Lim, "Through-Silicon-Via Aware Interconnect Prediction and Optimization for 3D Stacked ICs," Proc. of ACM/IEEE International Workshop on System Level Interconnect Prediction, pp. 85—92, 2009.

[9] J. Cong and Y. Zhang, "Thermal via planning for 3-D ICs," Proc. of IEEE/ACM International conference on Computer-aided Design, pp. 745—752, 2005.

[10] L. Jing and M. Hiroshi, "Post-placement Thermal Via Planning for 3D Integrated Circuit," Proc. of IEEE Asia Pacific Conference on Circuits and Systems, pp. 808—811, 2006.

[11] J. Cong and Y. Zhang, "Thermal-Driven Multilevel Routing for 3-D ICs," Proc. of IEEE/ACM Asia and South Pacific Design Automation Conference, pp. 121—126, 2005.

[12] V. Krishnan and S. Katkoori, "A 3D-Layout Aware Binding Algorithm for High-Level Synthesis of Three-Dimensional Integrated Circuits," Proc. of IEEE International Symposium on Quality Electronic Design, pp. 885—892, 2007.

[13] P.D. Robert, L.R. David, and W. Wayne, "Tgff: Task Graphs for Free," Proc. of IEEE International Workshop on Hardware/software Codesign, pp. 97—101, 1998.

[14] C.T. Hwang, J.H. Lee and Y.C. Hsu, "A Formal Approach to the Scheduling Problem in High Level Synthesis," IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 10, no. 4, pp. 464—475, 1991.

[15] C.A. Papachristou and H. Konuk, "A Linear Program Driven Scheduling and Allocation Method Followed by an Interconnect Optimization Algorithm", Proc. of IEEE/ACM Design Automation Conference, pp. 77—83, 1990.

[16] J. Ramanujam, S. Deshpande, J. Hong, and M. Kandemir, "A Heuristic for Clock Selection in High-Level Synthesis," Proc. of IEEE/ACM Asia and South Pacific Design Automation Conference, pp. 414—419, 2002.

[17] C.J. Tseng, R.W. Wei, S.G Rothweiler, M. Tong, and A.K. Bose, "Bridge: A Versatile Behavioral Synthesis System," Proc. of IEEE/ACM Design Automation Conference, pp. 415—420, 1988.

[18] T. Kim, J.W.S. Liu, and C.L. Liu, "A Scheduling Algorithm for Conditional Resource Sharing," Proc. of IEEE/ACM International Conference on Computer Aided Design, pp. 84—87, 1991.

[19] S.H. Huang and C.H. Cheng, "An ILP Approach to the Simultaneous Application of Operation Scheduling and Power Management", IEICE Transactions on Fundamentals of Electronics, Communications, and Computer Sciences, vol. E91-A, no. 1, pp. 375—382, 2008.