

# Chemical-Mechanical Polishing Aware Application-Specific 3D NoC Design

Wooyoung Jang, Ou He<sup>\*</sup>, Jae-Seok Yang<sup>†</sup>, and David Z. Pan<sup>‡</sup>

SoC Platform Development Team, Samsung Electronics, Yongin-City, South Korea

<sup>\*</sup>IBM System & Technology Group, Beijing, China

<sup>†</sup>CAE Team, Samsung Electronics, Hwaseung-City, South Korea

<sup>‡</sup>Department of Electrical and Computer Engineering, the University of Texas at Austin, Austin, USA  
wooyoung.jang@samsung.com, heou@cn.ibm.com, js.yang@samsung.com, dpan@ece.utexas.edu

## ABSTRACT

In this paper, we propose the first chemical-mechanical polishing (CMP) aware application-specific three-dimensional (3D) network-on-chip (NoC) design that minimizes through-silicon-via (TSV) height variation, thus reduces its bonding failure, and meanwhile optimizes conventional NoC design objectives. Our 3D NoC design assigns cores to proper silicon layers, determines the 3D NoC topology, allocates routing paths, and then floorplans cores, routers and TSV arrays by a CMP-aware manner. The key idea behind this 3D NoC design flow is to determine the CMP-aware 3D NoC topology where TSV arrays with low and uniform metal density are inserted between adjacent layers. Experimental results show that our CMP-aware 3D NoC design can achieve lower TSV height variation, higher performance and lower power consumption than the previous state-of-the-art 3D NoC designs.

## 1. INTRODUCTION

Network-on-chip (NoC) is an effective solution for on-chip communication in three-dimensional (3D) interconnections. However, the 3D NoC must meet not only application constraints, but also manufacturing constraints imposed by the 3D technologies. So far, many researchers have addressed the issues of 3D floorplanning and 3D NoC topology generation with the consideration of thermal hot spots. Besides the thermal and related thermal-mechanic stress effects [2], one particular challenge is that the wide range of the metal area by through-silicon-vias (TSVs) and landing pads increases non-uniform metal density, and thus results in the critical variation of wire thickness and TSV height during a chemical-mechanical polishing (CMP) process [8]. The CMP processes can be used for the removal of extra Cu on silicon after filling TSVs with Cu or depositing Cu on TSV landing pads, called Cu-CMP and silicon backside thinning, called silicon-CMP. The uneven Cu-wire thickness changes wire resistance and coupling capacitance between wires, and thus results in critical timing variation. Moreover, the uneven TSV height leads to bonding failure between TSVs and landing pads. To mitigate the non-uniform metal density, dummy metal or TSV can be filled in empty spaces, but it may affect RC parasitics [8][9] and significantly reduce usable silicon area.

In this paper, we propose the first CMP-aware application-specific 3D NoC design that minimizes TSV height variation, thus reduces bonding failure, and meanwhile optimizes conventional NoC design objectives. For NoC vertical links composed of tens to hundreds of TSVs, the layout of each individual TSV is not efficient since it results in complex global routing and TSV manufacturing stress affects more transistors [2]. Therefore, *TSVs should be placed as an array type in 3D NoC.*

The work is done while Wooyoung Jang was a Ph.D. student at the University of Texas at Austin.

However, the array with dense TSVs is sensitive to a CMP process which results in high TSV height variation, and thus leads to bonding failure. Moreover, if the arrays with different TSV density are used in the same layer, bonding TSVs on landing pads is more difficult. Therefore, *TSVs in an array should be placed with a pitch resulting in low CMP variation endured by a bonding technique and TSV arrays with the same density should be inserted in each layer.* In addition, since the size of the TSV arrays is too large, *TSV arrays should be handled during the floorplanning stage in physical design.* Based on these motivations, the major contributions of this paper include:

- 1) We show that TSV height variation during silicon-CMP process is more severe in 3D NoC.
- 2) We propose a CMP-aware application-specific 3D NoC design which consists of core-to-layer assignment, topology synthesis, and floorplanning.
- 3) We show that the proposed 3D NoC design reduces TSV height variation with lower design cost, and meanwhile achieves less hop count, wirelength, and power consumption.

The rest of this paper is organized as follows: Section 2 reviews related works. Section 3 introduces CMP and Cu-Cu thermo-compression direct bonding, and then addresses various TSV layouts and their CMP variation. Section 4 formulates CMP-aware application-specific 3D NoC design problems. Section 5 presents detailed techniques of the proposed algorithms. Section 6 shows experiment results and Section 7 concludes this paper.

## 2. RELATED WORKS

Several works for 3D ICs have explored thermal floorplanning [3][6]. A fast thermal-driven floorplanning algorithm with a 3D floorplan representation and integrated compact resistive network thermal model was proposed in [3]. Hung et al. presented interconnect and thermal-aware floorplanning for 3D microprocessors in [6]. Recently, researchers have synthesized a 3D NoC topology with the limited number of TSVs. Yan et al. presented a 3D NoC synthesis algorithm that made use of accurate power and delay models for 3D wiring with TSVs [14]. In [10] and [12], 3D NoC topology synthesis algorithms based on a direct extension of the 2D NoC synthesis procedure was proposed.

However, these previous 3D NoC designs have not considered CMP variation which may lead to bonding failures and timing variations. In addition, as routers and TSV arrays are becoming as large as other cores, the previous 3D NoC designs would suffer low physical design quality. This is because 3D floorplanning was first performed without the large routers and the TSV arrays, and thus there might be not enough spaces to place the routers and TSV arrays [10][12]. Even if 3D floorplanning is again performed after deciding a 3D NoC topology [14], their design qualities such as wirelength, power consumption, and area are severely limited.

### 3. PRELIMINARIES

#### 3.1 Chemical-Mechanical Polishing Process

One of the most potential sources of yield loss and timing variation in 3D technologies is TSV bonding on land pads. In a typical industrial bonding procedure [13], a TSV-wafer is ground down to a target thickness slightly above the TSV depth (keeping TSVs unexposed) and further thinned using a CMP process. The chemical reaction creates a hydroxylated-form material which has weaker atomic bonds. Then, a mechanical surface abrasion aided by slurry particles removes the material. Fig. 1(a) shows the uneven surface of wafer backside after grinding and CMP. Subsequently, the polished silicon surface is plasma-etched, such that the TSVs protrude from the wafer as shown in Fig. 1(b). On the contrary, TSV Landing pads are commonly fabricated on the top metal layer in a damascene process and designed to be larger than TSVs to prevent overlay error. The top metal layer with land pads is also polished by CMP to remove overburden Cu. Finally, a wafer or die with landing pads is bonded with a different wafer or die with TSVs.

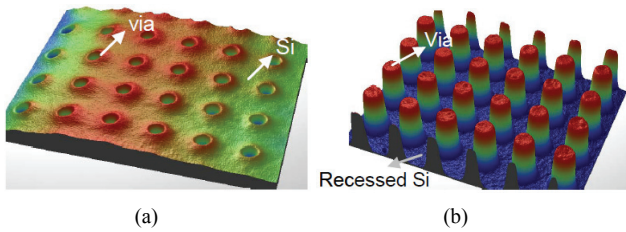


Fig. 1: Local topography on backside of wafer after (a) grinding and CMP and (b) Si-recess etch following CMP [13].

#### 3.2 TSV Layouts and TSV Height Variation

Silicon-CMP is just used for finely thinning silicon after grinding silicon backside since the processing time of CMP is too long. As silicon-CMP involves simultaneous polishing of silicon, Cu, and barrier, their removal rates are different according to both different chemical effects on the materials and different TSV densities. The different removal rates of these materials results in different polish times across the wafer backside. For example, in Fig. 2(a), by the time the silicon and barrier under TSVs used for a 64-bit link are cleared at a point, the silicon and barrier under TSVs used for 128-bit links might have been not cleared yet. Hence, either the silicon and barrier under the 128-bit TSVs are underpolished at the time the silicon and barrier under the 64-bit TSVs are cleared or the 64-bit TSVs are overpolished at the time the silicon and barrier under the 128-bit TSVs are cleared. Fig. 2(a) shows the 128-bit TSVs are underpolished after silicon-CMP. In [13], IMEC TSV technology showed that within-die thickness variation after silicon-CMP was  $1.5\mu\text{m}$  for a die size of  $10.6 \times 10.6\text{mm}^2$  when TSVs of which the diameter, pitch, and density are  $5\mu\text{m}$ ,  $10\mu\text{m}$ , and  $10\text{k}/\text{mm}^2$ , respectively, were evenly distributed over the whole chip. The within-die thickness variation is sensitive to high and irregular TSV density and directly related to TSV height variation. Consequently, the uneven TSV height variation can induce TSV bonding failure as shown in Fig. 2(a). In particular, the bonding failure will be more severe in a Cu-Cu direct thermo-compression bonding technique since TSVs must be directly contacted to landing pads without any micro-bump. Metal fill synthesis is not an efficient solution for silicon-CMP since dummy TSV insertion would significantly increases the overall chip area.

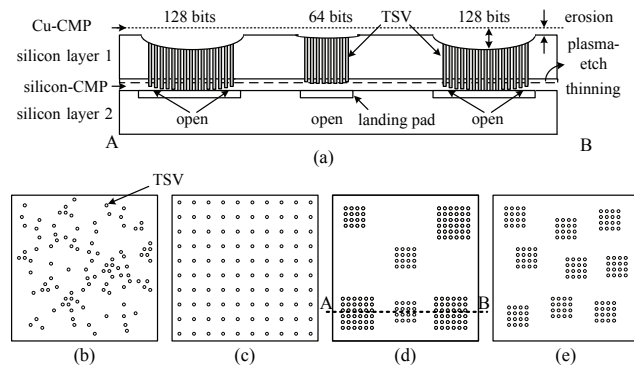


Fig. 2: TSV layouts and TSV height variation induced by CMP process.

TSVs can be placed with different schemes during placement and routing [2]. If TSVs are laid out without any constraints imposed by 3D technology, they can be distributed as shown in Fig. 2(b). Whereas such layout achieves much shorter wirelength, TSV height variation induced by silicon-CMP greatly increases due to uneven TSV density. In Fig. 2(c), TSVs are placed with globally uniform density distributions. The TSV distribution provides the least TSV height variation to 3D ICs. However, such TSV layout is not suitable for NoC vertical links composed of tens to hundreds of TSVs since it results in so complex global routing that any wire in the same vertical link may detour with a long path. The long wires detoured makes system performance degraded or timing closure difficult. In addition, the layout of each individual TSV causes manufacturing stresses to more devices [2]. Therefore, grouping TSVs to an array and then laying out the array is more desirable for 3D NoC.

In Fig. 2(d), there exist two kinds of TSV arrays. The small array includes a one-way link and the large array includes a two-way link which may have two times more TSVs than the one-way link. TSVs in the small array fail to contact landing pads since TSVs in the large array are less cleared than those in the small array during silicon-CMP such that the surface in a die is uneven. In Fig. 2(a) that is the cross section of AB in Fig. 2(d), the 64-bit TSV array has the strong possibility of failing to contact landing pads on silicon layer 2 since the 128-bit TSV array is underpolished. In addition, since the metal density of the 128-bit TSV is high, its own silicon-CMP variation can be so high that TSVs in the array have the possibility of failing to contact landing pads. We can control the local TSV density defined as the size of a TSV array divided by a TSV pitch. If the 128-bit TSV array has a wider TSV pitch, its density can be as low as that of the 64-bit TSV array. However, since it has the penalty of area, we focus on reducing the size of a TSV array as shown in Fig. 2(e).

### 4. PROBLEM FORMULATION

In most previous application-specific 3D NoC designs [10][12][14], 3D floorplanning is first performed and then a 3D topology is determined as shown in Fig. 3(a), where their 3D technology constraint is the number of allowable TSVs. However, there may be no enough dead space where routers and TSV arrays can be physically placed after deciding a 3D topology [10][12]. In order to prevent overlapping routers, TSV arrays, and cores already floorplanned, additional floorplanning is performed in each layer [14], but such 3D NoC design flow is not efficient for reducing wirelength, hop count, and thus energy consumption. Furthermore, the layout of TSV arrays without considering CMP variation leads to TSV bonding failure on landing pads.

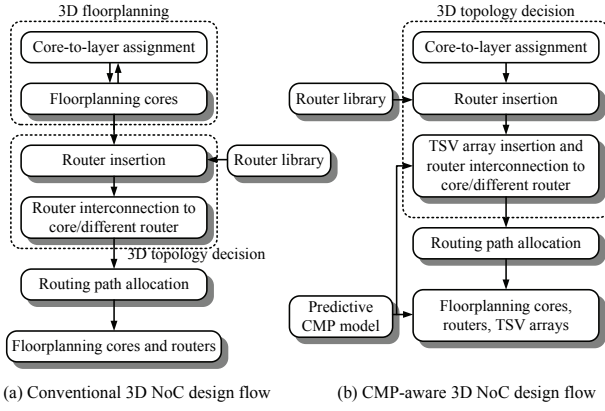


Fig. 3: Application-specific 3D NoC design flows.

Fig. 3(b) shows the proposed CMP-aware NoC design flow covering such issues. We first assign  $n$  cores to  $k$  layers with the purpose of using fewer TSVs. The number of allowable routers is inserted in each layer, and then the routers are interconnected to cores and different routers in the same layer, where the number of interconnecting any router to cores and other routers is given. Next, any routers are also interconnected to different routers in adjacent layers by only one-way vertical links. Since the number of allowable TSVs between layers is also limited, vertical interconnections are placed for the minimum hop count. Then, routing paths without deadlock and livelock are allocated on the existing interconnections. A TSV pitch for the one-way link is computed and a TSV array is composed. Finally, all cores, routers, and TSV arrays are simultaneously floorplanned in each layer.

We start to solve the CMP-aware application-specific 3D NoC issues from a core graph. A graph  $G(V, E)$  with  $n$  vertices is a directed graph, where each vertex  $v_i \in V$  represents a core, a router, and a TSV array and each directed edge  $e_{i,j} \in E$  represents communication relation between  $v_i$  and  $v_j$ .  $vol(e_{i,j})$  represents communication volume between  $v_i$  and  $v_j$  and  $wl(e_{i,j})$  represents wirelength between  $v_i$  and  $v_j$ .

#### 4.1 Core-to-Layer Assignment

Core-to-layer assignment allows cores to move from continuous space to discrete space, forcing each core to exactly occupy one layer. That is, a set of cores  $V = \{v_1, v_2, \dots, v_n\}$  is assigned to  $k$  layers  $L = \{l_1, l_2, \dots, l_k\}$ , and thus  $V = \{V^{l_1}, V^{l_2}, \dots, V^{l_k}\}$  is obtained, where  $V^{l_i} = \{v_1^{l_i}, v_2^{l_i}, \dots, v_j^{l_i}\}$ , where  $j < n$ . The area of cores is represented as  $\{A_1, A_2, \dots, A_n\}$ . To equally assign the area of cores to layers, an area constraint is defined as:

$$\alpha_{\min} \sum_{i=1}^n \frac{A_i}{k} < A_i < \alpha_{\max} \sum_{i=1}^n \frac{A_i}{k} \quad (1)$$

where  $\alpha_{\min}$  and  $\alpha_{\max}$  are acceptable minimum and maximum area coefficients ( $\alpha_{\min} < 1 < \alpha_{\max}$ ), respectively. We use the thermal model proposed in [3]. With the area constraint, the objective of our layer assignment is to minimize communication between layers and lower temperature as follows:

$$\min \left[ \beta_1 \sum vol(e_{i,j}) \cdot |u - v| + \beta_2 \left\{ \sum_{p=1}^k \left( P_p \sum_{q=1}^p R_q \right) + R_b \sum_{p=1}^k P_p \right\} \right] \quad (2)$$

s.t.  $\forall v_i \in V^{l_u}, \forall v_j \in V^{l_v} (u \neq v)$

where  $\beta_1$  and  $\beta_2$  are weighting coefficients,  $R_q$  is a thermal resistor in layer  $q$ ,  $P_p$  is the sum of current source in layer  $p$ , and  $R_b$  is the thermal resistor of the bottom layer material.

#### 4.2 3D NoC Topology Decision and Routing Path Allocation

Given the number of allowable routers and TSV arrays and the number of different routers interconnected to one router in each layer, we interconnect a router to cores and different routers in the same layer. A router communication graph  $RCG(R, C)$  with  $m$  vertices is a directed graph, where each vertex  $r_i \in R$  represents a router, and each directed edge  $c_{i,j} \in C$  represents communication between  $r_i$  and  $r_j$ . The objective of our 2D topology is as follows:

$$\min \left[ \sum vol(e_{i,j}) \cdot dist(M(v_i), M(v_j)) \right] \quad (3)$$

$$\text{s.t. } bw(link(M(v_i), M(v_j))) \geq vol(e_{i,j}), (\forall v_i, \forall v_j) \in V^{l_u}$$

where  $dist(r_p, r_q)$  is distance (hop count) between  $r_p$  and  $r_q$  and  $M()$  is a core-to-router mapping function.  $link(r_p, r_q)$  is all links which any packet in  $r_p$  passes for reaching  $r_q$ . Then, we interconnect routers in adjacent layers, based on the RCG graphs. The objective of our topology decision among layers is as follows:

$$\min \left[ \sum vol(e_{i,j}) \cdot dist(r_p, r_q) \right] \quad (4)$$

$$\text{s.t. } link_{TSV}(r_p, r_q) \neq link_{TSV}(r_q, r_p),$$

$$bw(link(r_p, r_q)) \geq vol(e_{i,j}), \forall v_i \in V^{l_u}, \forall v_j \in V^{l_v} (u \neq v)$$

where  $link_{TSV}(r_p, r_q) \in link(r_p, r_q)$  is a vertical link which any packet in  $r_p$  passes for reaching  $r_q$ . This equation indicates that routers in different layers are interconnected by only one-way vertical links. Thus, CMP variation can be greatly reduced and the yield of TSV bonding can be greatly improved.

#### 4.3 Floorplanning

Based on our predictive CMP model, we compute a TSV pitch where a used bonding technique must endure TSV height variation in the number of TSVs covering a one-way vertical link. Then TSV arrays are inserted between any routers in adjacent layers. As the inputs of our floorplanner, we take a set of cores, routers, and TSV arrays,  $\{v_1, v_2, \dots, v_n\}$ .  $v_i$  is a  $W_i \times H_i$  rectangle and aspect ratio  $H_i/W_i$ . Each block can be free to rotate and change the aspect ratio continuously in a given range  $[AR_{\min, i}, AR_{\max, i}]$ . A floorplan  $F$  is the assignment of  $(x_i, y_i)$  for each block  $v_i$  without any overlap of all cores, routers and TSV arrays, where half-perimeter wire length estimation is used. We use the thermal model proposed in [3], which minimizes the maximum temperature difference in the same layer. Finally, the objective of our floorplan  $F$  is as follows:

$$\min \left[ \gamma_1 \sum A_i + \gamma_2 \sum (wl(e_{p,q}) \times vol(e_{p,q})) \right. \\ \left. + \gamma_3 (\max(T(x, y, l_u)) - \min(T(x, y, l_u))) \right] \quad (5)$$

$$\text{s.t. } (\forall v_i, \forall v_p, \forall v_q) \in V^{l_u}, \max(wl(e_{p,q})) < th_w$$

where  $\gamma_1$ ,  $\gamma_2$ , and  $\gamma_3$  are weighting factors.  $T(x, y, l_u)$  is the temperature of a tile in  $x$ ,  $y$ , and  $l_u$  at  $x$ -axis,  $y$ -axis, and layer, respectively and  $th_w$  is the maximum allowable wirelength.

## 5. CMP-AWARE 3D NOC DESIGN

### 5.1 CMP-Aware Core-to-Layer Assignment

Since the number of TSVs required depends on communication volume between different layers, the communication volume should be minimized together with thermal consideration. In addition, the area of each layer should meet the area constraint, Eq. (1). Fig. 4 shows two different core-to-layer assignment approaches where eight cores are assigned to four layers. Let a core graph given as shown in Fig. 4(a) where all edges have the same weight, all cores have the same power density, and the number is the area of a core for simple explanation.

The first approach is that 4-way minimum-cut area-balanced partitioning is performed, and then the partitioned subgroups are one-to-one assigned to different layers. For example, in Fig. 4(b), the cores are partitioned to {A, B}, {C, D}, {F, G}, and {E, H} that have the same area and the minimum cuts. Then, the partitioned subgroups are one-to-one assigned to any layers, achieving the minimum hops as shown in Fig. 4(c).

The second approach we propose in Algorithm 1 recursively performs area-balanced bi-partitioning with the minimum cost computed from Eq. (2). Fig. 4(d) shows the result of the first bi-partitioning where the same area and the minimum cut are obtained (line 2). Then, any core which communicates other cores in a different layer is assigned in advance, depending on their communication gain as shown in Fig. 4(e) (line 5). The communication gain is computed as the subtraction of the amount of intra-layer communication from that of inter-layer communication. If the communication gain of any core is greater or equal to 0, the core is assigned to a current layer. In Fig. 4(e), core B, C, E, and F communicate cores in a different layer and their communication gains are 0, -1, 0, and 0, respectively. Thus, core B, E, and F are assigned to boundary layers. Then, the second bi-partitioning in each sub-group is again performed for the minimum cut under the area constraint. Fig. 4(f) shows the final result where hop count between layers is 7 whereas the first approach is 8. Therefore, the second one can require fewer TSVs. The basic idea of Algorithm 1 can be easily extended even if the number of a given layer is not a power of two.

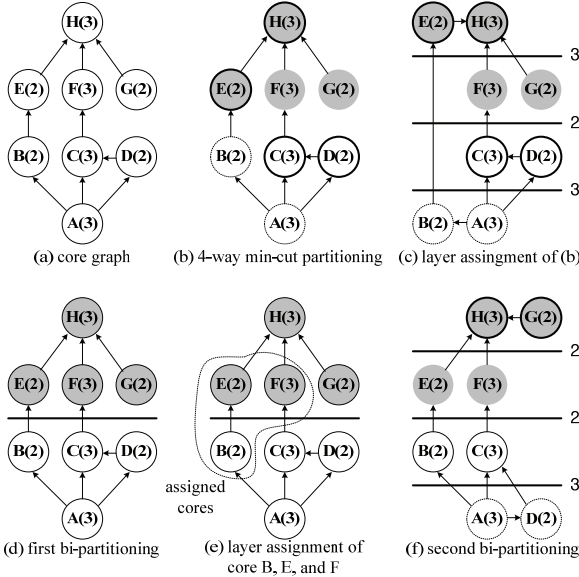


Fig. 4: Examples of assigning eight cores to four layers.

#### Algorithm 1: Core-to-Layer Assignment by Recursive Bi-Partitioning

```

1: while the number of partitioned layers is not equal to the target
   number of layers do
2:   Find bi-partitions of cores with min. cost computed by Eq. (2);
3:   Compute communication gain ( $CG_i$ ) of core  $i$  in layer  $k$ ;
4:   if  $CG_i \geq 0$  then
5:     Core  $i$  is assigned to layer  $k$ ;
6:   end if
7: end while

```

### 5.2 CMP-Aware 3D NoC Topology Decision

Since a 3D network topology decision problem is NP-Hard, we present efficient heuristics in this section. Furthermore, since the integrated problem makes it difficult to reach guaranteed quality bounds on the solution, we divide the 3D network topology decision problem into two distinct subproblems, called router-to-core/router interconnection in the same layer and router-to-router interconnection between different layers, and then we solve the respective subproblems. Whereas a bandwidth requirement can be easily satisfied by finding alternative routing paths or adding more interconnection resources, satisfying latency constraints is difficult if cores communicating each other are too wide apart. Therefore, any master core sensitive to latency should be interconnected to the same router as its slave core. A TSV array covering a one-way vertical link is used for interconnection between different layers and any router is not interconnected to routers in a different layer if it is already interconnected to the router with one direction as shown in Eq. (4), which minimizes TSV density variation, thus reduces TSV height variation resulting in TSV bonding failure.

#### 5.2.1 2D Router-to-router/core interconnection

Given a core graph, the number of allowable routers ( $max\_router$ ), and the number of allowable interconnection to a router ( $max\_int$ ), our 2D topology synthesis technique interconnects possible cores to any routers. The objective of our 2D topology decision is to minimize power consumption in each layer. Varying the number of routers in NoC designs has a great impact on power consumption and communication latency. NoC using few routers leads to longer core-to-router interconnections and hence, higher interconnection power consumption. On the contrary, when a number of routers are used, data flows have to traverse more routers, leading to high router power consumption and increasing area. Thus, we need to explore NoC designs with the different number of routers to obtain the best solution, starting from a design point where each core is interconnected to the minimum routers to one where cores are connected to the maximum allowable routers ( $max\_router$ ) in each layer.

The objective of Algorithm 2 is to establish efficient physical links between a router and a router/core in each layer. First,  $i$ -way minimum-cut partitioning is performed for cores in the same layer under the  $max\_int$  constraint (line 2) and then each group is assigned to one router (line 3). Next, links between the routers are inserted according to user's design objective (line 4). We implement the minimum spanning tree (MST) or point-to-point (P2P) interconnection. MST first interconnects two vertices nearby. Similarly, since two routers,  $r_p$  and  $r_q$  which heavily communicates each other should be interconnected with high priority, we use  $1/vol(c_{p,q})$  as the distance information. Then, the breadth-first-search or depth-first algorithms are used for searching MST. Next, a new router communication graph (RCG) is generated and then a prohibited turn set for the RCG is build to



**Algorithm 2: Topology Decision within Layer**

```

1: for  $i = \text{max\_router}$  to (the number of core/max_int) do
2:   Find  $i$ -way min-cut partitions under  $\text{max\_int}$  constraints;
3:   Assign each group to one router;
4:   Interconnect routers by user's design objective;
5:   Build router communication graph (RCG);
6:   Build prohibited turn set for RCG to avoid deadlocks;
7:   Find paths for flows across different routers in each layer;
8:   Evaluate latency and bandwidth;
9:   Go to line 2 if application constraints are not satisfied;
10: end for
11: Choose the best topology and design point;

```

avoid deadlocks (line 5-6). Based on the links, paths for flows across different routers in the same layer are allocated, using Dijkstra's shortest path algorithm (line 7). Application constraints such as communication latency and bandwidth are evaluated (line 8). If they are not satisfied, a different topology for the layer is again synthesized (line 9). Finally, the best topology and design point are selected among all generated topologies (line 11).

### 5.2.2 Layer-to-layer interconnection

After deciding a 2D topology in all layers, any layers must be interconnected to adjacent layers. Since we use the minimum TSV arrays, total hop count may increase according to the location of the TSV arrays. In addition, inserting either both one-way and two-way links in the same layer or a TSV array with high metal density results in severe TSV height variation during CMP. Thus, the objective of our layer-to-layer interconnection is to insert one-way links between layers for locally uniform TSV distribution and the minimum hop count under performance constraints. In our technique, if a one-way vertical link for  $c_{p,q}$  is established, the opposite one-way link for  $c_{q,p}$  is removed in the list of TSV array insertion candidates, where  $r_p \in V^m$  and  $r_q \in V^n$ , ( $m \neq n$ ).

For example, Fig. 5(a) is a core graph assigned to two layers, where the weight of all edges is 1. After deciding its topology in each layer, we can insert TSV arrays for the minimum hop count as shown in Fig. 5(b) and (c). TSV height variation during CMP is critical in the two-way link with high metal density. Therefore, Fig. 5(c) is desirable for low and uniform local TSV density.

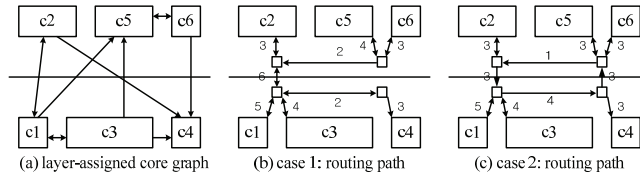


Fig. 5. CMP-aware router-to-router interconnection in adjacent layers.

### 5.3 CMP-Aware Floorplanning

We first compute a TSV pitch for one-way links, based on our predictive CMP model. The pitch must result in low TSV height variation endured by a bonding technique. Then, a TSV array is build and then simultaneously floorplanned with routers and cores in each layer. The goal of our floorplanning is to generate the layout that minimizes area, power consumption, and peak temperature. We modify an existing floorplanning technique [5] and invoke it with our unique cost function.

The power consumption on the given network can be presented as the power required by physical links. It is desirable to place cores, routers, and TSV arrays nearby if they heavily communicate. This is because the power consumption is directly proportional to the number of hop and the length of link. Hence,

we defined the cost function as the product of communication volume  $vol(e_{i,j})$  and wirelength  $w_{i,j}$  in Eq. (5). In addition, it is necessary to place cores, routers, and TSV arrays communicating within the allowable wirelength.

## 6. EXPERIMENTAL RESULTS

### 6.1 TSV Density and Predictive CMP Model

Fig. 6 shows TSV heights measured from the latest 3D ICs of IMEC after silicon-CMP, where the TSV diameter is  $5\mu\text{m}$  [7]. With these industry measurement data, we model TSV height variation as follows:

$$hv = 0.8017 \ln\left(\frac{s}{p}\right) + 1.226 \quad (6)$$

where  $hv$  is TSV height variation,  $s$  is the size of TSV array, and  $p$  is a TSV pitch in the array. Based on this model, we can compute a TSV pitch for the size of a given TSV array, which guarantees TSV height variation endured by a bonding technique. For example, if the size of a TSV array including a one-way link (113 wires) for OCP is  $11 \times 11$ , its TSV pitch must be at least  $14.58\mu\text{m}$  for less than  $1\mu\text{m}$  TSV height variation. On the contrary, if the size of a TSV array including a two-way link is  $16 \times 16$ , its TSV pitch must be at least  $21.21\mu\text{m}$ . Thus, their areas are  $0.0256\text{mm}^2$  and  $0.1151\text{mm}^2$ , respectively. Consequently, two one-way vertical links can show lower CMP variation or smaller design area than a single two-way vertical link.

### 6.2 CMP-Aware Application-Specific 3D NoC

We implement the CMP-aware application-specific (CAS) 3D NoC and [14] on GSRC Benchmarks with 100, 200 and 300 modules [4]. Wafers are stacked in a face-to-back fashion and we set a diameter and pitch of TSV to  $5\mu\text{m}$  and  $10\mu\text{m}$ , respectively.

Table 1 shows TSV height variation when various network interfaces are used. The local TSV density of CAS is more uniform and lower than that of [14] and CAS has 17.9% lower TSV height variation than [14]. Using only one-way links results in increasing hop count since it may not provide the shortest path. However, our 3D NoC design flow recovers the penalty of the hop count and even improves total hop count since a topology

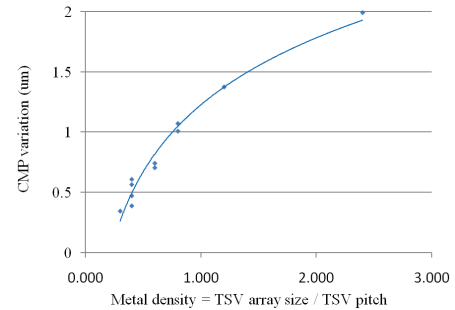


Fig. 6. Silicon-CMP variation based on IMEC wafer measurement [7].

Table 1: TSV Height Variation Comparison ( $\mu\text{m}$ ).

Network protocol	# of wire of one(two)-way link	[14]	CAS	Imp. (%)
AHB [1]	137 (274)	1.651	1.372	16.9
AXI [1]	204 (408)	1.821	1.551	14.8
APB [1]	99 (198)	1.551	1.226	21.0
OCP [11]	113 (226)	1.603	1.302	18.7
Average		1.657	1.363	17.9

decision is first performed.

Table 2 shows total hop count. CAS achieves, on average, 15% lower hop count than [14]. CAS tends to further improve hop count in complex NoC with a number of modules and layers. In addition, when a network is synthesized with limited resources like MST, CAS further improves hop count. As shown in Table 3, CAS achieves just 0.3% longer total wirelength than [14] in MST and even 4.6% shorter total wirelength than [14] in P2P.

Table 2: Hop Count Comparison.

the number of layer		4	5	6	7	8	I.(%)	
M	n100	[14]	1410	1470	1625	1671	1927	16.5
		CAS	1201	1254	1299	1361	1650	
	n200	[14]	3341	3366	3459	3737	3927	20.0
		CAS	2654	2931	2698	2934	3043	
	n300	[14]	5211	5158	5178	5257	5230	22.6
		CAS	4065	3912	4077	3984	4118	
Imp. (%)		20.5	19.0	21.3	22.4	20.5	<b>19.7</b>	
P	n100	[14]	1193	1336	1488	1629	1799	13.1
		CAS	1077	1194	1207	1416	1575	
	n200	[14]	2051	2487	2744	3136	3285	8.7
		CAS	2041	2278	2441	2788	2968	
	n300	[14]	2638	3279	3433	4163	4371	11.1
		CAS	2626	2943	3405	3234	3695	
Imp. (%)		2.3	9.7	8.0	16.7	12.9	<b>11.0</b>	

Table 3: Total Wirelength Comparison (mm).

the number of layer		4	5	6	7	8	I.(%)	
M	n100	[14]	9.6	8.4	7.5	7.1	6.5	-0.3
		CAS	9.6	8.5	7.4	7.1	6.6	
	n200	[14]	22.6	19.1	16.6	15.1	13.8	0.0
		CAS	23.2	19.1	16.6	15.0	13.3	
	n300	[14]	46.5	39.5	34.4	30.5	27.2	-0.6
		CAS	47.0	40.0	34.1	30.7	27.3	
Imp. (%)		-1.4	-0.9	0.7	-0.2	0.6	<b>-0.3</b>	
P	n100	[14]	47.2	38.6	32.9	29.6	26.4	2.1
		CAS	46.5	38.3	32.3	28.0	26.2	
	n200	[14]	95.4	77.7	67.8	61.3	55.2	4.7
		CAS	89.6	75.1	65.0	58.5	52.3	
	n300	[14]	144.6	129.9	115.5	102.6	94.5	7.0
		CAS	132.1	119.6	108.6	99.4	86.1	
Imp. (%)		7.5	5.4	4.8	3.9	6.5	<b>4.6</b>	

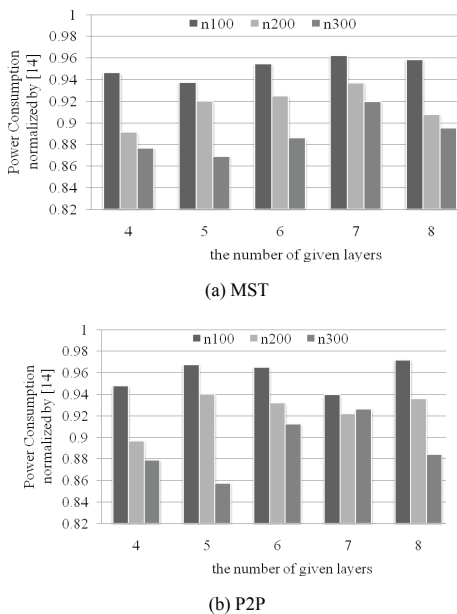


Fig. 7: Power consumption normalized by [14].

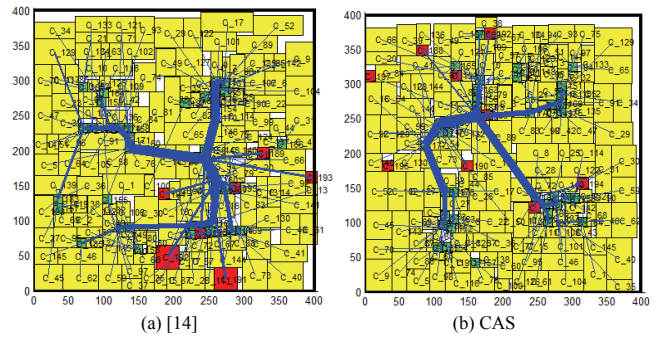


Fig. 8: The layout of application-specific 3D NoC designs

Fig. 7 shows power consumption normalized by [14]. The power consumption of CAS is 8.1% and 7.8% lower than that of [14] in MST and P2P, respectively. The total area of CAS is slightly smaller than [14] since CAS has smaller total TSV array area than [14]. The runtime of CAS ranges from 48-99 seconds in n300, which is about three times faster than [14].

Fig. 8 show the layouts generated by [14]+MST and CAS+MST, where blue lines show communication relations and their thickness indicates communication volume. Yellow rectangles, red rectangles, and green rectangles are cores, TSV arrays, and routers, respectively. Whereas Fig. 8(a) includes both one-way and two-way links, Fig. 8(b) includes just one-way links. Therefore, TSV heights are less variable, thus TSVs can directly contact landing pads easily.

## 7. CONCLUSION

In this paper, we proposed the first CMP-aware application-specific 3D NoC design. Our vertical integration managing architecture, physical design, and manufacturing issues together enables a reliable and robust 3D NoC with low power consumption and high performance. In particular, our CMP-aware 3D NoC approach reduces TSV height variation during the CMP process, and thus prevents bonding failures and timing variation.

## 8. REFERENCES

- [1] AMBA AXI, AHB, and APB Protocol Specifications. ARM [online]. Available: <http://www.arm.com>
- [2] K. Athikulwongse et al., "Stress-Driven 3D-IC Placement with TSV Keep-Out Zone and Regularity Study," in *Proc. International Conference on Computer-Aided Design*, 2010.
- [3] J. Cong et al., "A thermal-driven floorplanning algorithm," in *Proc. International Conference on Computer-Aided Design*, 2004.
- [4] GSRC Floorplan Benchmarks [Online]. Available: <http://vlsicad.eecs.umich.edu/BK/GSRCbench>.
- [5] O. He et al., "A novel fixed-outline floorplanner with zero dead space for hierarchical design," in *Proc. International Conference on Computer-Aided Design*, 2008.
- [6] W.-L. Hung et al., "Interconnect and thermal-aware floorplanning for 3D microprocessors," in *Proc. International Symposium on Quality Electronic Design*, 2006.
- [7] IMEC [Online]. Available: <http://www.imec.be>.
- [8] A. B. Kahng and K. Samadi, "CMP fill synthesis: a survey of recent studies," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 27, no. 1, pp. 3-19, 2008.
- [9] D. H. Kim et al., "Enabling 3D integration through optimal topograph," in *Proc. International Workshop on Design for Manufacturability and Yield Workshop*, 2010.
- [10] S. Murali et al., "Synthesis of networks on chips for 3D systems on chips," in *Proc. Asia South Pacific Design Automation Conference*, 2009.
- [11] Open Core Protocol Specification. OCP-IP [Online]. Available: <http://www.ocpip.org>.
- [12] C. Seiculescu et al., "SunFloor 3D: A tool for networks on chip topology synthesis for 3D systems on chips," in *Proc. Design, Automation, and Test in Europe*, Mar. 2009.
- [13] B. Swinnen et al., "3D integration by Cu-Cu thermo-compression bonding of extremely thinned bulk-Si die containing 10um pitch through-Si vias," in *Proc. International Electron Devices Meeting*, 2006.
- [14] S. Yan and B. Lin, "Design of application-specific 3D networks-on-chip architectures," in *Proc. International Conference on Computer Design*, 2008.