

Whitespaces-Aware TSV Arrangement in 3-D Clock Tree Synthesis

Wulong Liu, *Student Member, IEEE*, Yu Wang, *Senior Member, IEEE*, Guoqing Chen, *Member, IEEE*, Yuchun Ma, *Member, IEEE*, Yuan Xie, *Member, IEEE*, and Huazhong Yang, *Senior Member, IEEE*

Abstract—Through-silicon-via (TSV) could provide vertical connections among different dies in 3-D integrated circuits (3-D ICs), but the significant silicon area occupied by TSVs may bring great challenge to designers in 3-D clock tree synthesis (CTS), because only a few whitespace blocks can be used for clock TSV insertion after floorplan and placement are determined, specifically in the area-efficient 3-D IC designs. This paper proposes a whitespace-aware TSV arrangement algorithm in 3-D CTS, which mainly consists of three stages: sink preclustering, whitespace-aware 3-D method of means and medians (3-D-MMMs) topology generation, and deferred-merge embedding merging segment reconstruction. By leveraging the TSV-to-TSV coupling model, we also propose an efficient clock TSV arrangement method to alleviate the coupling effect of adjacent TSVs. Compared with the traditional 3-D-MMM-based CTS with TSV moving adjustment, the experimental results show that our proposed algorithm is more practical and efficient, achieving 49.2% reduction on the average skew and 1.9% reduction on the average power.

Index Terms—3-D integrated circuits (3-D ICs), clock tree synthesis (CTS), through-silicon-via (TSV) arrangement, whitespace.

I. INTRODUCTION

WITH CMOS process technology continuously scaling down, through-silicon-via (TSV)-based 3-D integrated circuits (3-D ICs) have drawn much more attention recently. With the help of 3-D technology we can reduce global wirelength, alleviate congestion, and improve performance. Moreover, 3-D technology provides much more design flexibility by heterogeneous integration [1].

Manuscript received January 15, 2014; revised May 26, 2014 and July 23, 2014; accepted August 10, 2014. Date of publication September 18, 2014; date of current version August 21, 2015. This work was supported in part by the 973 Project under Grant 2013CB329000; in part by the National Science and Technology Major Project under Grant 2010ZX01030-001-001-04; in part by the National Natural Science Foundation of China under Grants 61373026, 61261160501, and 61028006; and in part by the Tsinghua University Initiative Scientific Research Program.

W. Liu, Y. Wang, and H. Yang are with the Department of Electrical Engineering, Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: wulong.liu@gmail.com; yu-wang@tsinghua.edu.cn; yanghz@tsinghua.edu.cn).

G. Chen is with the AMD China Research Laboratory, Beijing 100190, China (e-mail: guoqing1.chen@amd.com).

Y. Ma is with the Department of Computer Science, Tsinghua University, Beijing 100084, China (e-mail: myc@tsinghua.edu.cn).

Y. Xie is with the Department of Electrical and Computer Engineering, University of California at Santa Barbara, Santa Barbara, CA 93106, USA (email: yuanxie@gmail.com).

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/TVLSI.2014.2354347

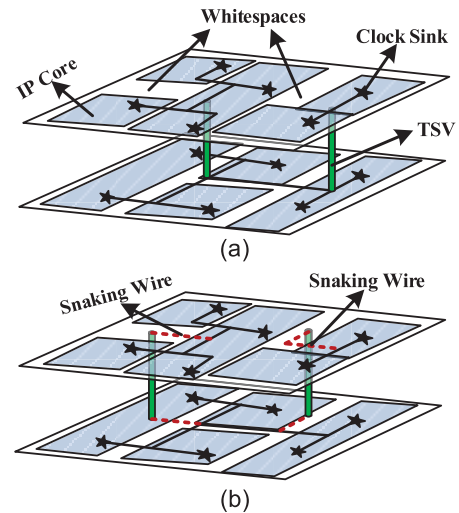


Fig. 1. Three-dimensional CTS without whitespace-aware TSV arrangement. (a) TSVs are not located in whitespace after an initial design. (b) Moving TSVs into whitespace incurs longer wirelength and leads to potential skew increase.

For a 3-D stacked IC, the clock network distributes the clock signal through the entire stacks and connects all the clock sinks on different dies by a single tree as shown in Fig. 1. Different from the 2-D clock network, the clock signal is distributed not only through x and y directions, but also in z direction through TSVs, which increases the design complexity. Despite the obvious superiority of 3-D ICs, the vertical interconnect, TSV could also lead to some serious problems, such as the limited whitespace for TSV-insertion and the relatively severe parasitic or coupling effect of TSVs.

Under current technologies, TSVs are very huge compared with gates and memory cells [2], therefore, a large number of TSVs will consume significant silicon area and degrade the yield and reliability of the chip. Furthermore, as TSVs are usually placed in the whitespace between macroblocks or cells, a bad arrangement of TSVs may incur longer wirelength because the available TSV location might be far away from its connected cells. Nowadays, intellectual property (IP) and standard cell-based design has been extensively used to reduce design cost; however, only a few whitespace blocks are reserved for clock TSVs after floorplan and placement are determined [3]. Fig. 1 indicates that without the consideration of TSV whitespace during 3-D clock tree synthesis (CTS), TSV moving is necessary to ensure that each

TSV is located inside the whitespace and it would incur longer wirelength and lead to potential skew increase. In addition, the parasitic and coupling effects of TSVs located in the limited whitespace blocks could be very problematic owing to the big sizes of TSVs, which may aggravate the path delay and power consumption, and may also lead to timing violations. Therefore, the impact of TSVs should be carefully considered in the design of 3-D clock network. In this paper, we mainly focus on optimizing the TSV arrangement in the limited whitespace.

A. Previous Work

Different from the 2-D clock network, the main challenge in 3-D clock network is to alleviate the negative impacts of TSV and the vertical stacking processing on different design criteria, such as reducing the power consumption, enhancing the performance (e.g., skew and slew), increasing the robustness under thermal and process variations, and ensuring the prebond testability. Many literatures spring up in the past few years in the field of 3-D CTS, which mainly focus on zero (bounded) skew [4]–[6], low power [5], [7]–[9], robustness [10]–[13], and prebond testability [8], [9], [14]–[16].

In one of the most representative methods, Minz *et al.* [17] generated a 3-D clock tree considering the number of TSVs by defining a TSV bound between adjacent dies in their 3-D method of means and medians (3-D-MMMs) algorithm. The basic idea is to recursively divide the given sink set into two subsets until each sink belongs to its own set. The division is based on the TSV bound, which is also divided according to the ratio of the estimated number of TSVs in each subset. The 3-D-MMM-ext algorithm [7] gives the optimal number of TSVs to minimize the overall power consumption. Kim *et al.* proposed MMM-3-D algorithm [18], which uses a designer-specified parameter ρ ($0 \leq \rho \leq 1$) to control the partition direction. If the half-perimeter wirelength of a subset is smaller than ρL (where L is the half-perimeter wirelength of all the sinks), z -cut is executed. They also proposed a solution called ZCTE-3-D to solve the zero skew clock tree embedding problem, which can give the best TSV allocation and placement result for a given tree topology. These top-down methods could control TSV counts but are not able to accurately predict TSV locations.

In the aforementioned previous works, there is still little effort on solving the challenge induced by the large TSVs in the 3-D clock network. Zhao and Lim [19] solved a practical 3-D clock routing problem which considered the obstacles induced by different TSVs, such as P/G, signal, and clock TSVs. They developed a TSV-induced obstacle-aware deferred-merge embedding (DME) method to construct a buffered clock tree which can avoid those obstacles with the help of newly defined merging segments. In practice, besides the TSV-induced obstacles, the IP-based designs may also lead to many other obstacles to prevent the TSV insertion. Generally, only a few whitespace blocks are reserved for clock TSVs after floorplan and placement are determined in IP and standard cell-based designs. Long wire detour is inevitable in such scenarios. Taking the available whitespace blocks

rather than the obstacles as the constraints can reduce the design complexity and enhance the performance. Thus, a novel whitespace-aware 3-D CTS algorithm is necessary.

Another issue in the previous works is that the TSVs are only simplified as 2C-R [7], [17], [18], [20] model, which underrates the impact of TSVs on the 3-D clock network. Meanwhile, fruitful work has been done to model the parasitic and coupling effects of TSVs, such as [21]–[26] focusing on the TSV-to-TSV coupling effects in device or full-chip level, and [27] focusing on the TSV to active circuit coupling effect. In digital 3-D ICs, the TSV-to-TSV coupling effect is much more significant, which may lead to timing violations and extra power consumption. However, little work has been conducted to evaluate the coupling effect of adjacent TSVs when constructing the 3-D clock network, and it is a challenging task to build a high-performance 3-D clock network while alleviating the TSV-to-TSV coupling effect in the limited whitespace blocks.

B. Our Contribution

As mentioned before, the number and locations of TSVs are crucial and only a few whitespace blocks are available for clock TSVs during 3-D CTS. None of the existing methods still works efficiently in this scenario. In this paper, we propose a whitespace-aware TSV arrangement algorithm in 3-D CTS. The main contributions are summarized as follows.

- 1) We formulate the whitespace-aware TSV arrangement problem in 3-D CTS and propose a practical and efficient algorithm to solve the problem. Furthermore, we propose a whitespace-aware 3-D CTS flow in Section III.
- 2) The proposed algorithm is made up of three stages: first, a distance-aware sink preclustering algorithm, which distributes the sinks to nearby whitespace blocks; second, an extended version of the 3-D-MMM clock tree topology generation algorithm named as TSV whitespace-aware 3-D-MMM (TWA-3-D-MMM for short), which ensures that each sink set contains whitespace blocks; and third, a DME merging segment reconstruction algorithm, which brings convenience to routing and TSV arrangement.
- 3) Unlike previous 3-D CTS methods which simplify the TSV as a 2C-R model, in this paper, we leverage the TSV-to-TSV coupling model to evaluate the TSV parasitic/coupling effects, and propose an efficient clock TSV arrangement method to alleviate the TSV coupling effects.
- 4) We investigate the relation between whitespace area, TSV number, and the main CTS quality criteria such as power, skew, and slew rate by comparing our method with the traditional 3-D-MMM based CTS with TSV moving adjustment. We apply our method to the main-stream ISPD benchmarks and real industry cases; the experimental results show the superiority of our method, which can achieve an average skew and power reduction of 49.2% and 1.9%, respectively.

The rest of this paper is organized as follows. Section II presents the preliminaries and problem formulation

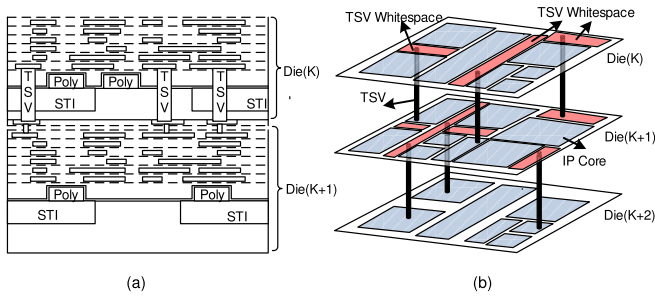


Fig. 2. Models. (a) F2B stack. (b) TSV between Die(k) and Die($k + 1$) is only restricted by the whitespace blocks on Die(k).

of 3-D CTS. Section III illustrates the detailed algorithms of our proposed whitespace-aware 3-D CTS. Our experimental setup and experimental results are presented in Section IV. Finally, we summarize the work in Section V.

II. PRELIMINARIES AND PROBLEM FORMULATION

A. Electrical Model of 3-D Clock Network

Die: For a N -die stacked 3-D clock design, we number the dies as Die(0), Die(1), \dots , Die($N - 1$) in a top-down manner, the die on which the clock source is located is named as the source die. For simplicity, we set the clock source on Die(0) in this paper.

TSV: TSV between nonadjacent dies is composed of several TSVs between adjacent dies. In this paper, we model the TSVs with the TSV-to-TSV coupling effect. The detailed coupling model between two adjacent TSVs is presented in Section II-B.

TSV Whitespace Block: With current technologies, the diameter of TSV is very huge compared with gates and memory cells, therefore only a few whitespace blocks can be reserved for TSVs before CTS. TSV whitespace blocks exist between IP blocks, and they can be modeled as discrete whitespace blocks. In a N -die face-to-back (F2B) stack case as shown in Fig. 2, TSVs between Die(k) and Die($k + 1$) are only restricted by the whitespace blocks on Die(k) [28]. Note that TSV whitespace on the last die, that is, Die($N - 1$) is useless. For simplicity, TSV whitespace (blocks) is referred to as whitespace (blocks) hereafter.

B. TSV-to-TSV Coupling Model

In 3-D ICs, the coupling effect between two adjacent TSVs could be significant because of the big sizes of TSVs. This TSV-to-TSV coupling could lead to extra delay or power, and timing violations. In this paper, we adopt the simplified equivalent lumped model of two coupled TSVs [23] to evaluate the impact of TSVs on the 3-D clock network. The model of two coupled TSVs is shown in Fig. 3. We use the following simplified formulas to calculate the capacitances and the resistances:

$$C_{TSV} = \frac{1}{4} \frac{2\pi \epsilon_0 \epsilon_r}{\ln \left(\frac{r_{TSV} + t_{OX}}{r_{TSV}} \right)} \times l_{TSV} \quad (1)$$

$$C_{Si} = \epsilon_0 \epsilon_{si} \frac{2(r_{TSV} + t_{OX}) + \alpha}{d} \times l_{TSV} \quad (2)$$

$$C_{Bump} = \frac{\epsilon_0 \epsilon_r}{d - 2r_{Bump}} \times \pi \times r_{Bump} \times l_{Bump} \quad (3)$$

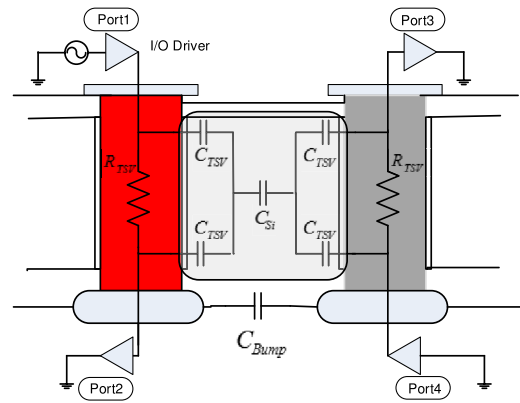


Fig. 3. TSV-to-TSV coupling model.

$$R_{TSV} = \frac{l_{TSV}}{\sigma \pi r_{TSV}^2} \quad (4)$$

$$R_{Si} = \frac{\epsilon_{si}}{C_{si} \sigma} \quad (5)$$

where ϵ_0 and ϵ_{si} are the dielectric constant of vacuum and silicon, α is the scaling factor, r_{TSV} and l_{TSV} are the TSV radius and height, r_{Bump} and l_{Bump} are the radius and the height of a bump, t_{OX} is the thickness of the insulator, and d is the distance between two TSVs. To explore the latency induced by TSV coupling effect, we apply a pulse signal to one TSV and treat the other TSV as victim, then simulate the equivalent circuit model in SPICE with the parameters defined by Chaabouni *et al.* [25]. The simulation result shows that the latency through a TSV can be reduced by 65%, if the distance to adjacent TSV is increased from 11 to 100 μm . This TSV-to-TSV coupling-induced latency uncertainty may induce timing violations in 3-D digital ICs.

C. Problem Formulation

The formal definition of whitespace-aware TSV arrangement problem in 3-D CTS is as follows. Given some whitespace blocks W , a set of clock sinks S , a TSV bound B_{TSV} , and a slew rate bound B_{Slew} , the objective is to construct a single clock tree such that: 1) the number of clock TSVs, that is, $TSV_{Num} \leq B_{TSV}$; 2) each clock TSV is located in the whitespace blocks without overlap; 3) clock slew rate is under B_{Slew} ; and 4) clock skew and clock power are minimized.

III. ALGORITHM

A. Overview of Our Proposed Method

Our proposed TSV whitespace-aware 3-D clock synthesis mainly consists of three stages: 1) sink preclustering; 2) TSV whitespace-aware 3-D-MMM clock tree topology generation; and 3) DME merging segment reconstruction stage. In the sink preclustering stage, sinks far away from their related whitespace are clustered to form subtrees, only the root node of the subtree is reserved and treated as a new sink. In the TWA-3-D-MMM clock tree topology generation stage, we extend the 3-D-MMM method by judging whether the current x/y -cut between multiple dies is appropriate such that each

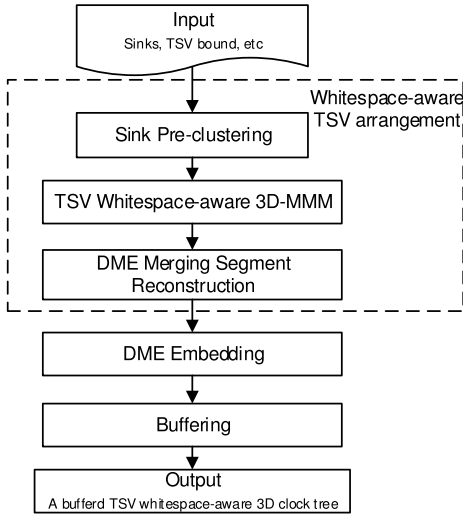


Fig. 4. Proposed whitespace-aware 3-D CTS flow.

sink set contains whitespace blocks. In the DME merging segment reconstruction stage, we modify the merging segment of the internal nodes having TSVs by considering TSV geometries and whitespace occupation, which would benefit detail routing and TSV arrangement. By integrating a slew-aware buffering stage, we further present a whitespace-aware 3-D CTS flow in Fig. 4. The computational complexity of our proposed method is $O(mn)$ where n and m are the number of clock sinks and number of whitespace blocks, respectively.

B. Sink Preclustering

Because the reserved whitespace blocks for TSV insertion are relatively narrow and small, and the clock sinks are widely distributed, there may be a long distance between sinks and whitespace blocks. Ignoring the available whitespace blocks during 3-D clock network and then moving the TSVs into the whitespace would lead to wirelength overhead and potential skew increase.

To solve this problem, an intuitive method is to distribute sink nodes closer to the whitespace, which is called sink preclustering. The preclustering algorithm proposed in this paper is shown in Fig. 5. First, we put all whitespace blocks from different dies on a plane and name it as a whitespace set. Second, for each die, we calculate the minimal distance from each sink to the whitespace set through an exhaustive search and assign the sinks to their nearest whitespace blocks. Third, we use a designer-specified parameter β to control sink preclustering. For each die, sinks that have a longer distance from their related whitespace block than the value βL (where L is the half-perimeter wirelength of the die) need to be clustered. For each sink cluster, we generate a subtree by using the classical method of means and medians (MMMs) [29] and DME [30] for clock tree topology generation and detail routing. The root of the subtree is treated as a new sink with its latency and downstream capacitance as input delay and capacitive load, while all the original sinks in the cluster are removed from the sink set. After preclustering, the sink set that contains nonclustered sinks and cluster roots is set as the new constraint to construct the whole 3-D clock network.

C. TSV Whitespace-Aware 3-D-MMM

The basic idea of the famous 3-D-MMM algorithm is to recursively divide the given sink set and related TSV bound into two subsets until each sink belongs to its own set. TSVs are necessary when merging nodes on different dies. The algorithm tends to use as many TSVs as the giving bound permits, but in terms of whitespace, this division may cause serious problems. In Fig. 6(a), under current y -cut, sink $s1$ and $s2$ from different dies are divided into a subset with no whitespace in it, so a TSV is inserted and moved into the nearest whitespace, which leads to longer wirelength. To deal with this problem, we modify the 3-D-MMM algorithm and extend it to the TWA-3-D-MMM algorithm by judging whether the current x/y -cut between multiple dies is appropriate considering whitespace. The pseudocode of the proposed TWA-3-D-MMM method is shown in Fig. 7. In line 2, we initialize the subsets $S1$ and $S2$. In lines 3 and 4, if the current sink set contains only one node, which means it is a sink itself, then return. If not, we execute x/y -cut and divide the current sink set and TSV bound into two subsets when sinks in the current set are on different dies. Then, we come to the most important judging procedure (line 11) in our algorithm.

Assuming sink set S is divided into two subsets $S1\{s_{11}, s_{12}, \dots, s_{1i}\}$ and $S2\{s_{21}, s_{22}, \dots, s_{2j}\}$ under current x/y -cut, and the maximum and minimum die number of sinks in $S1$ and $S2$ are $d_{\max1}, d_{\min1}, d_{\max2}, d_{\min2}$, respectively. In multiple die case, $d_{\max1} \neq d_{\min1}$ and $d_{\max2} \neq d_{\min2}$. For subset $S1$, all sinks have to be connected, which means TSVs are needed between adjacent dies from $\text{Die}(d_{\min1})$ to $\text{Die}(d_{\max1})$, so subset $S1$ should contain whitespace on $\text{Die}(d_{\min1}), \text{Die}(d_{\min1} + 1), \dots, \text{Die}(d_{\max1} - 1)$, and so should subset $S2$. If one of the subsets does not meet the whitespace constraints, the current cut is canceled and marked to be z -cut, which usually happens near the leaf level of the clock tree. Fig. 6 shows a judging example.

- 1) When executing current y -cut, there is no whitespace in sink subsets $\{s1, s2\}$, so the TSV-related parent node a of sinks $s1, s2$ is initially arranged outside the whitespace and should be moved to the nearest whitespace, which would incur longer wirelength and lead to potential skew increase.
- 2) Because there is no whitespace in sink subsets $\{s1, s2\}$, we change current cut to z -cut, so the TSV is arranged into whitespace without longer wirelength.
- 3) When judging current y -cut, subset $S1$ has no whitespace in $\text{Die}(k + 1)$, so current cut is canceled and changes to z -cut.
- 4) Both subset $S1$ and $S2$ has whitespace in $\text{Die}(k)$ and $\text{Die}(k + 1)$, so current cut is valid.

D. DME Merging Segment Reconstruction

There are two phases in the classical DME clock routing method: 1) a bottom-up phase computes all feasible locations for the roots of recursively merged subtrees, saved as related merging segments; and 2) a top-down phase then resolves the exact embedding of these internal nodes [30]. For those internal nodes with TSVs, their related merging segments need

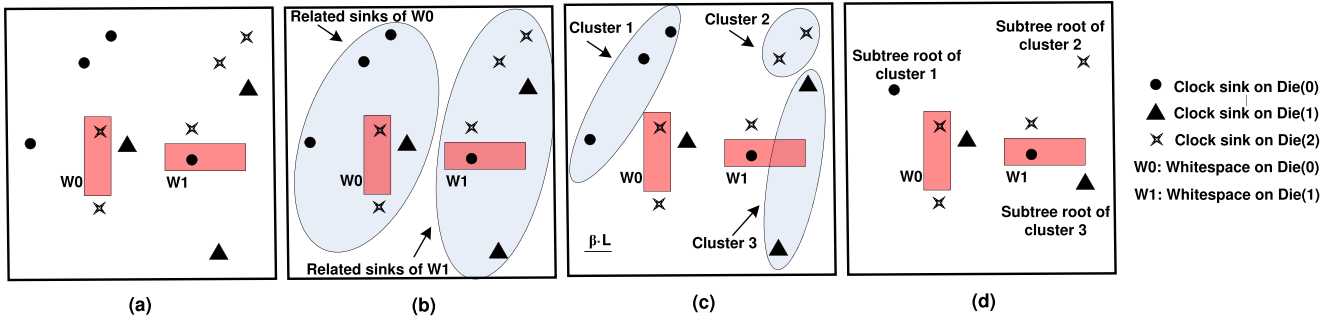


Fig. 5. Sink preclustering illustration. (a) Before preclustering. (b) Arranging sinks to their related whitespace blocks according to distance. (c) For each whitespace block, generate clusters for its related sinks those are more than βL far away from the whitespace block and on the same die. (d) Subtree roots of the clusters are treated as new sinks while other nodes in the cluster are neglected.

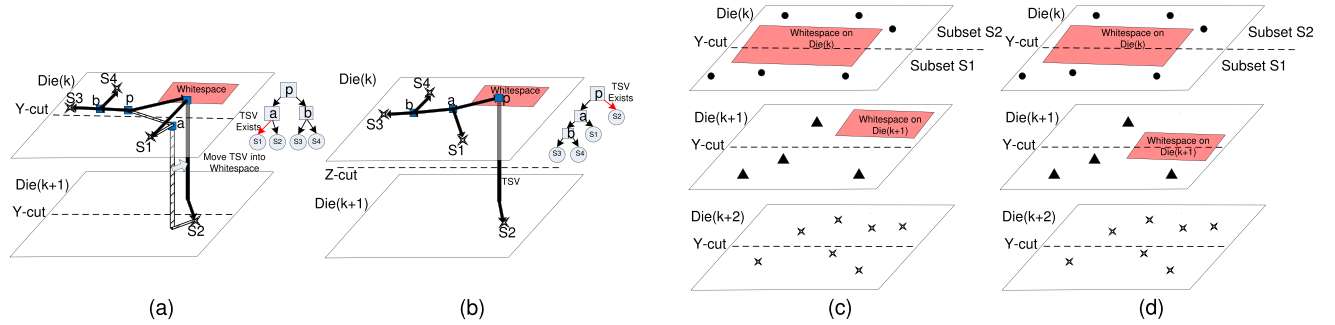


Fig. 6. Example to compare the traditional 3-D-MMM and our TWA-3-D-MMM clock tree topology generation methods. (a) Traditional 3-D-MMM with TSV moving. (b) Our TWA-3-D-MMM method. (c) and (d) Two different cases in our TWA-3-D-MMM clock tree topology generation method.

TSV Whitespace-aware 3D-MMM Topology Generation (TWA-3D-MMM)

Input: clock sinks, TSV bound, TSV whitespace, cutDirection

Output: a rooted 3D clock tree topology

```

1: TWA-3D-MMM (sinkset S, TSV bound B, Whitespace blocks W, cutDirection C)
2:   S1 and S2 = subset of S;
3:   if (|S| = 1) then
4:     return root(S);
5:   else if (B != 1 or stack(S) = 1) then
6:     if (C = x-cut) then
7:       x-cut(S, S1, S2);
8:       C = y-cut;
9:       Find B1, B2, such that B1 + B2 = B;
10:    if (C = y-cut) then
11:      y-cut(S, S1, S2);
12:      C = x-cut;
13:      Find B1, B2, such that B1 + B2 = B;
14:    if (B != 1) then
15:      if (there is no W in S1 or S2) then
16:        cancel current cut;
17:        C = z-cut;
18:        B = 1;
19:      if (B = 1 and stack(S) > 1) then
20:        z-cut(S, S1, S2);
21:        B1 = B2 = 1;
22:    root(S1) = TWA-3D-MMM(S1, B1, C);
23:    root(S2) = TWA-3D-MMM(S2, B2, C);
24:    leftChild(root(S)) = root(S1);
25:    rightChild(root(S)) = root(S2);
26:    return root(S);

```

Fig. 7. Pseudocode of our TWA-3-D-MMM.

to be reconstructed and settled into whitespace. In this paper, by leveraging the previously discussed TSV-to-TSV coupling model in Section II, we propose a method to alleviate this coupling effect of adjacent TSVs when arranging TSVs into the available whitespace.

The TSV-to-TSV coupling effect would be much more problematic, if there is voltage difference between the signals on two adjacent TSVs. If signals on adjacent TSVs are in-phase, the effective coupling capacitance (C_{si} in Fig. 3) is zero, resulting in a smaller latency through the TSVs. If signals on adjacent TSVs are out-phase, the effective coupling capacitance C_{si} is nonzero, which would result in glitches and delay variations in the signals, increasing the power consumption. For the clock network of 3-D ICs, we find that the out-of-phase coupling scenario mainly exists between adjacent clock TSVs at different clock tree levels. Fig. 8 shows a simple example to illustrate this effect. TSV3 and TSV4 are at the first level of clock network, whereas TSV2 and TSV1 are at the second and third levels of clock network, respectively. As shown in Fig. 8(c), because of the different arrival time at each clock TSV, there will be voltage difference between these clock TSVs for a portion of the clock cycle. Fig. 8(b) shows that the TSV-to-TSV coupling effect, which is directly related to the voltage difference between these adjacent TSVs, is also proportional to the tree level difference of these clock TSVs. For example, by using our proposed TSV whitespace-aware 3-D CTS method, TSV1, TSV3, and TSV4 are assigned into one whitespace block as shown in Fig. 8(a). To construct a low skew and balanced 3-D clock network, the distance between TSV1 and TSV3 (or TSV4) should be carefully designed.

With the consideration of TSV geometries, the available whitespace blocks, and the coupling effect of adjacent TSVs, we propose a TSV arrangement method in whitespace blocks to alleviate the noise and power consumption of 3-D clock network. First, we divide the whitespace into many small squares according to the TSV keepout zone as shown in

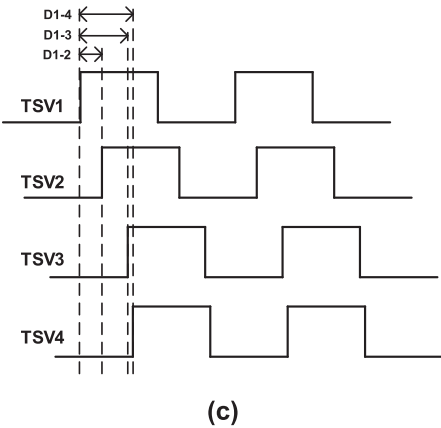
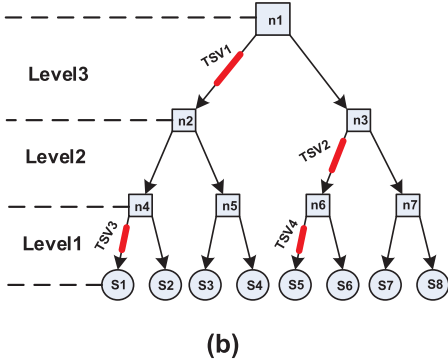
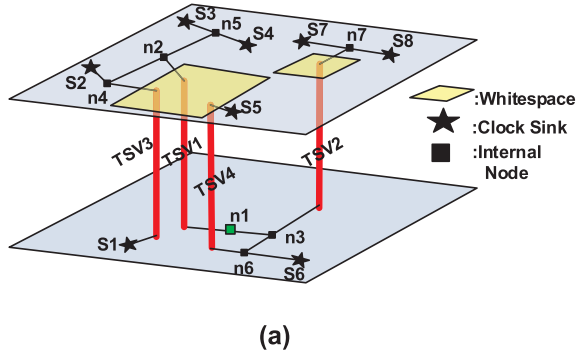


Fig. 8. An example to consider the coupling effect of adjacent TSVs in TSV arrangement. (a) Clock sinks distribution in a two-die F2B stacked chip. (b) The hierarchy of 3D clock tree. (c) The timing waveform for each TSV.

Fig. 9. Then, for those internal nodes with TSVs, their related merging segments need to be reconstructed and settled in whitespace. We identify the available whitespace square which has the smallest distance to the merging segment of the internal node with TSV, and use the center of that whitespace square as the temporary TSV location. All of the neighbor whitespace squares are checked to see whether it has been occupied by a TSV which causes large tree-level difference with the present TSV. If such scenario happens, the initially selected whitespace square for TSV insertion is abandoned, and the whitespace square with the second smallest distance to the merging segment is checked with the same procedure until finding the proper location for the internal node with TSV. Note that reconstructing the merging segment of one child node may induce imbalanced latency between two child nodes with the same parent node, which needs wire-snaking to balance the latency.

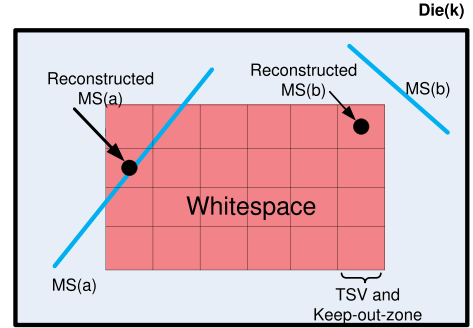


Fig. 9. DME merging segment reconstruction.

The center of the selected whitespace square is set as the new merging segment, and the delay and downstream capacitance of this segment are updated. This TSV movement would lead to certain wirelength increase; however, with the help of sink preclustering and TWA-3-D-MMM, merging segments will be close to whitespace, minimizing the impact of TSV moving. Once a whitespace square is used, it is marked as occupied. After merging segment reconstruction, we can execute the DME top-down embedding and generate the clock routing result.

E. Slew-Aware Buffering

Clock slew rate control is of great importance for high-speed clock design, because a large clock slew rate may cause extra power consumption and potential timing violations. To ensure the clock signal slew rate, we add a buffering stage to our whitespace-aware 3-D CTS flow. Two kinds of buffers are inserted: clock buffers and TSV-buffers [9]. Clock buffers are inserted along the wire to control latency and slew rate, whereas TSV-buffers are inserted just at each internal node for prebond testability. Different from existing 3-D designs, which focus on slew-aware buffer insertion during the bottom-up embedding procedure of DME [7], [9], [31], our slew-aware buffering is performed after clock routing, because it is easy to achieve with an $O(n)$ computational complexity. In our slew-aware buffering algorithm, clock buffers are added along the clock paths so that the downstream capacitance of each buffer is limited to the bounding condition, which is denoted as CMAX in [7]. Long snaking wire paths also need to be buffered. After initial buffer insertion, we insert redundant buffers at the sink node to make sure the buffer numbers from clock source to sinks are balanced. Then, we reduce the buffer number in a bottom-up merging method, that is, two buffers at each child node could be replaced with one buffer at the parent node.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. Experimental Setup

We implement our proposed method by using C++ programming language on Linux environment with 3 GHz processor and 4 GB memory. We use ISPD 2009 clock network synthesis contest benchmark [32] and two-die stacking for simplicity. In our experiments, we use technology parameters based on the 45-nm predictive technology model [33]. The parasitic resistance and capacitance of unit wire length

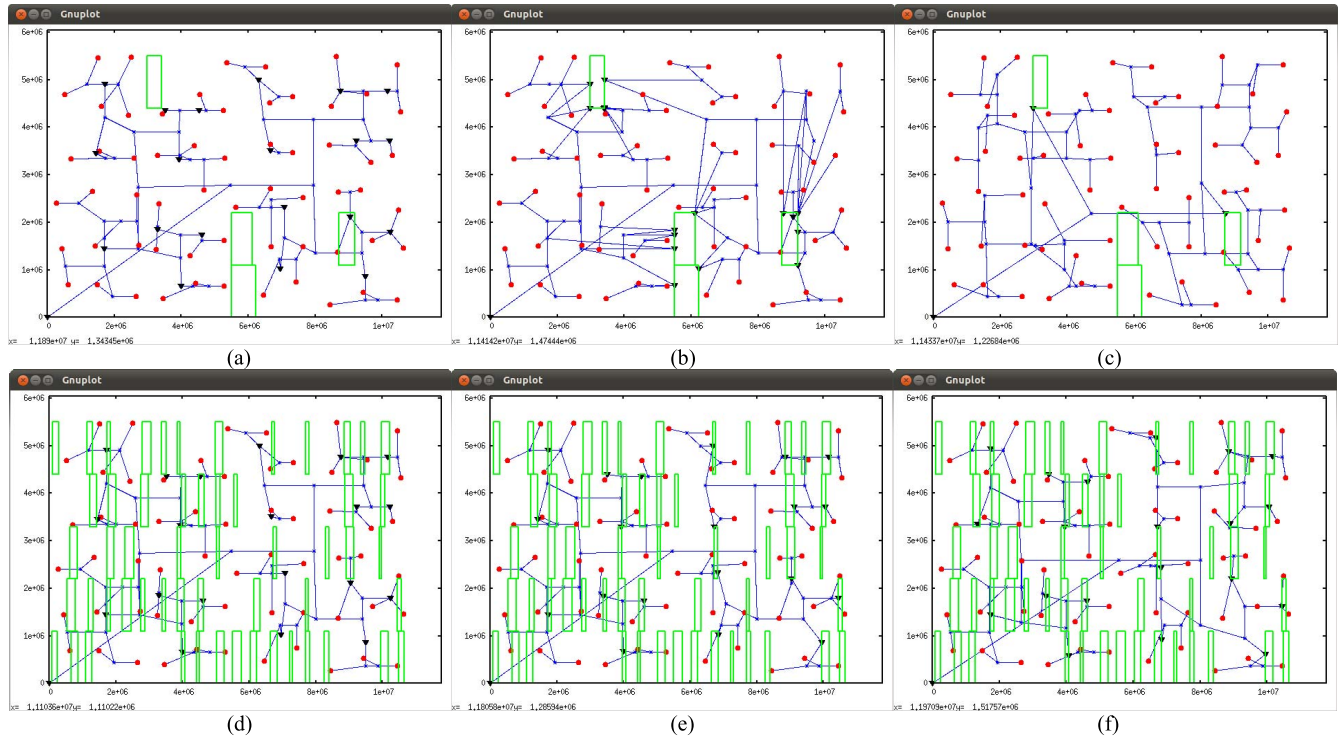


Fig. 10. Clock solution under different whitespace area. Red points and black triangles: sinks and TSVs, respectively. Green rectangles: whitespace blocks. (a) Number of blocks = 4, 3-D-MMM-DBM solution before TSV moving. (b) Number of blocks = 4, 3-D-MMM-DBM solution after TSV moving, with longer wirelength. (c) Number of blocks = 4, ours. (d) Number of blocks = 55, 3-D-MMM-DBM solution before TSV moving. (e) Number of blocks = 55, 3-D-MMM-DBM solution after TSV moving, with longer wirelength. (f) Number of blocks = 55, ours.

are $0.1 \Omega/\mu\text{m}$ and $0.2 \text{ fF}/\mu\text{m}$, respectively. The parameters of the TSV-to-TSV coupling model shown in Fig. 3 are referred to [25]. The TSV diameter with keepout zone is defined as $7.41 \mu\text{m}$ [19]. The buffer parameters are defined as the input capacitance is 35 fF ; the output capacitance and resistance are 80 fF and 61.2Ω , respectively. Because these benchmarks are originally designed for 2-D ICs, similar with previous work [7], [17], we divide these benchmarks into two layers and whitespace blocks are randomly generated between sinks. In addition, the clock frequency is set as 2 GHz , and the supply voltage is 1.2 V . Note that the runtime of our algorithm is within seconds for all benchmarks.

In SPICE simulation [34], wires are segmented with π model and TSVs are modeled as shown in Fig. 3. Clock slew rate is defined as the transition time from 10% to 90% of clock signal at each sink and buffer input. The clock slew rate requirement is 100 ps . The total wirelength of 3-D clock network can be calculated through our proposed algorithm, whereas the power consumption, clock skew, and clock slew are evaluated with SPICE simulation. The unit of wirelength, power, skew, and slew are reported in mm, W, ps, and ps, respectively.

B. Result Analysis

1) *Impact of TSV Whitespace Area:* We construct and simulate the entire 3-D clock tree by our proposed method on benchmark ispd09f11. To explore the impact of TSV whitespace on 3-D clock network, we widely change the number and area of the whitespace blocks, as shown in Fig. 10. Alternatively, we also implement the solution based on

3-D-MMM, DME routing, and buffering algorithm, which is named as 3-D-MMM-DBM hereafter. To deal with situations that internal nodes with TSVs are not arranged in the whitespace blocks, we simply move these internal nodes with their related TSVs into the nearest whitespace block, which may significantly increase the wirelength.

In Table I, it can be observed that the 3-D-MMM-DBM method is strongly influenced by the number and the area of the whitespace blocks. When fewer whitespace blocks are allowed, such as those shown in Fig. 10(a) and (b), TSVs have to be moved for a long distance. Although the performance of the 3-D-MMM-DBM is relatively good before TSV moving, moving TSVs into the whitespace blocks leads to extra power and increased skew, and also causes slew violations. The long wirelength induced by TSV moving, however, can be significantly reduced when whitespace blocks are widely distributed over the whole die as shown in Fig. 10(d) and (e), because there are more choices for TSV arrangement. Our proposed 3-D CTS solution tends to arrange each TSV in the whitespace blocks as expected, as shown in Fig. 10(c) and (f), resulting in better skew/slew/power, especially for scenarios with fewer and smaller whitespace blocks (which are more practical), as shown in Table I.

2) *Exhaustive Search Results for TSV Bound:* To explore the impact of TSV bound on 3-D clock network, we exhaustively sweep the TSV bound from 1 to 50 for the ispd09f11 benchmark with 16 whitespaces. As shown in Fig. 11, with the increase of TSV bound, the traditional 3-D-MMM-DBM solution suffers from severe power and skew problems, whereas our method shows consistent good results. This behavior

TABLE I
IMPACT OF DIFFERENT WHITESPACE AREA ON THE NUMBER OF TSV, SKEW, POWER, AND SLEW BETWEEN 3-D-MMM-DBM METHOD AND OUR PROPOSED METHOD (TSV BOUND IS SET TO BE 20, $BlockNum$ AND TSV_{Num} ARE THE NUMBER OF WHITESPACE BLOCKS AND TSVs, VIO IS THE SLEW VIOLATION)

| $BlockNum$ (area%) | 3D-MMM-DBM before TSV moving | | | | 3D-MMM-DBM after TSV moving | | | | Our method | | | |
|-----------------------|------------------------------|--------------|--------------|-------------|-----------------------------|--------------|--------------|-------------|---------------|--------------|--------------|-------------|
| | TSV Number | Skew (ps) | Power (W) | Slew Vio | TSV Number | Skew (ps) | Power (W) | Slew Vio | TSV Number | Skew (ps) | Power (W) | Slew Vio |
| 4(4.11%) | 20 | 23.805 | 0.299 | N | 20 | 175.784 | 0.358 | Y | 2 | 28.575 | 0.294 | N |
| 9(5.64%) | 20 | 23.805 | 0.299 | N | 20 | 148.983 | 0.336 | Y | 9 | 41.282 | 0.314 | N |
| 16(10.03%) | 20 | 23.805 | 0.299 | N | 20 | 55.135 | 0.314 | N | 14 | 40.088 | 0.308 | N |
| 29(12.86%) | 20 | 23.805 | 0.299 | N | 20 | 83.057 | 0.309 | N | 18 | 32.448 | 0.309 | N |
| 36(14.47%) | 20 | 23.805 | 0.299 | N | 20 | 40.938 | 0.304 | N | 19 | 26.410 | 0.310 | N |
| 55(16.23%) | 20 | 23.805 | 0.299 | N | 20 | 33.221 | 0.302 | N | 20 | 32.013 | 0.302 | N |
| 131(19.79%) | 20 | 23.805 | 0.299 | N | 20 | 22.977 | 0.301 | N | 20 | 25.334 | 0.302 | N |

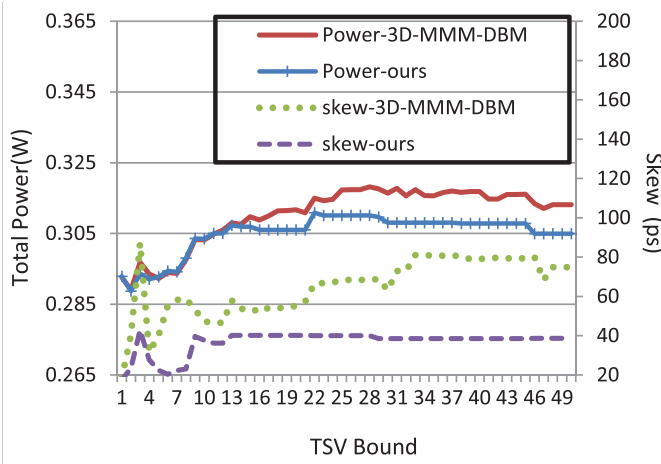


Fig. 11. Skew and power trends for ispd09f11 with different TSV bounds [1, 50] for both 3-D-MMM-DBM and our method.

happens because a larger TSV bound means more TSV moving adjustments, which may worsen the unbalanced clock latency.

We also implement our proposed whitespace-aware 3-D CTS method with different whitespace area as shown in Table I and sweep TSV bound in a much larger range from 1 to 130 to explore the impact of TSV bound and whitespace area on the power consumption and skew. An ideal case with unlimited whitespaces, which means the TSV can be placed anywhere, is defined as the baseline. As shown in Fig. 12, in most cases, the power consumption is decreased with the increase of TSV bound. The power consumption is also decreased with more whitespaces, because more whitespaces provide more flexibility for the TSV placement. Meanwhile, the skew is also improved with more whitespaces as shown in Fig. 13. Note that in real 3-D IC designs, although reserving more whitespaces for clock TSV insertion tends to improve the skew and power consumption, the induced area overhead should be carefully evaluated.

3) β of Preclustering: As illustrated in Section II, β plays an important role in cluster generation. Actually, there exists a β_{max} beyond which preclustering is meaningless. This phenomenon is easy to understand because when β is sufficiently large, none of the sinks needs to be clustered. We can find the longest distance from the sinks to their related whitespace blocks and calculate β_{max} . A sweeping

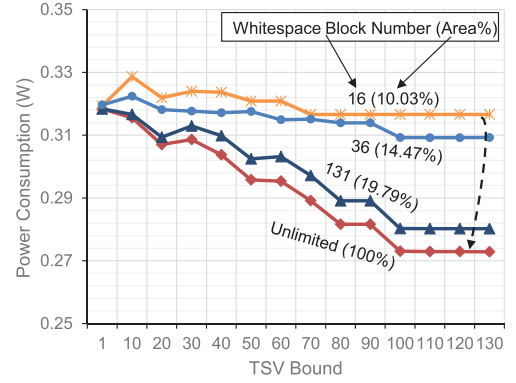


Fig. 12. Power consumption with different TSV bounds [1, 130] and with different whitespace area for our proposed whitespace-aware 3-D CTS method.

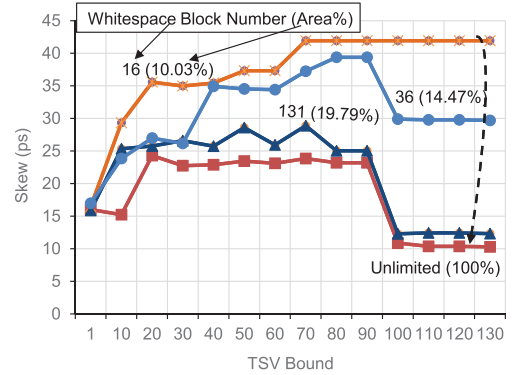


Fig. 13. Skew with different TSV bounds [1, 130] and with different whitespace area for our proposed whitespace-aware 3-D CTS method.

result shown in Fig. 14 reveals that the preclustering should be implemented carefully because a bad choice of β would unnecessarily cluster too many sinks, and affect topology and routing results. Practically, β in the range from 90% to 99% of the β_{max} provides appropriate results.

4) Wirelength, Skew, and Power Results: To fully explore the comparison of our method with 3-D-MMM-DBM, much more cases are examined with other benchmarks in ISPD09 contest [32], as shown in Table II. In all cases, the whitespace area is set to be around 10% of the whole die area with more than ten whitespace blocks. The results shown in Table II demonstrate that our method has no slew violations while

TABLE II
IMPACT OF DIFFERENT TSV BOUND ON DIFFERENT BENCHMARKS BETWEEN 3-D-MMM-DBM AND OUR METHOD

| Benchmark | Block _{Num} and Area (%) | TSV Bound | 3D-MMM-DBM | | | | Our method | | | |
|-----------|-----------------------------------|-----------|------------|-----------|----------|-----------------|------------|-----------|----------|-----------------|
| | | | skew (ps) | Power (W) | Slew Vio | Wirelength (mm) | Skew (ps) | Power (W) | Slew Vio | Wirelength (mm) |
| ispd09f11 | 16 (10.57%) | 1 | 18.4 | 0.295 | N | 185.09 | 17.5 | 0.295 | N | 185.37 |
| | | 10 | 47.2 | 0.305 | N | 171.98 | 37.6 | 0.305 | N | 174.81 |
| | | 20 | 55.1 | 0.314 | N | 171.34 | 40.1 | 0.308 | N | 169.76 |
| ispd09f12 | 15 (9.66%) | 1 | 21.9 | 0.279 | N | 164.44 | 21.9 | 0.279 | N | 164.93 |
| | | 10 | 78.8 | 0.286 | N | 150.74 | 27.0 | 0.282 | N | 157.87 |
| | | 20 | 63.9 | 0.291 | N | 196.44 | 32.3 | 0.285 | N | 158.17 |
| ispd09f21 | 15 (9.97%) | 1 | 26.4 | 0.299 | N | 199.86 | 27.2 | 0.300 | N | 199.43 |
| | | 10 | 149 | 0.308 | Y | 196.78 | 48.1 | 0.295 | N | 193.42 |
| | | 20 | 196 | 0.341 | Y | 208.49 | 46.4 | 0.302 | N | 191.47 |
| ispd09f22 | 12 (7.36%) | 1 | 29.9 | 0.238 | N | 132.56 | 25.4 | 0.238 | N | 133.01 |
| | | 10 | 70.5 | 0.229 | N | 121.52 | 40.2 | 0.233 | N | 126.32 |
| | | 20 | 67.6 | 0.239 | N | 140.39 | 55.1 | 0.239 | N | 126.21 |
| Average | / | / | 68.7 | 0.286 | / | 169.97 | 34.9 | 0.280 | / | 165.06 |

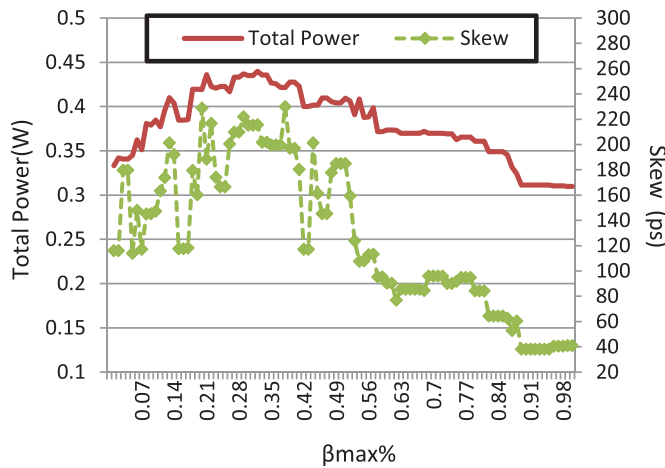


Fig. 14. Clock skew and power trends for ispd09f11 based on different β values: from 0% to 100% of the β_{max} .

3-D-MMM-DBM does. Meanwhile, our method achieves an average skew reduction of 49.2%, an average power reduction of 1.9%, and an average wirelength reduction of 2.9%, respectively. Because all the TSVs must be restricted to the whitespace blocks, the unavoidable longer wires aggravate the clock skew, while our method can minimize the skew degradation and reduce the wirelength, slew violations, and power consumption. Note that although only two-layer stacked case is implemented for simplicity, our proposed whitespace-aware 3-D CTS method can be applied for cases with more stacked layers.

5) *Analysis of the TSV-to-TSV Coupling in 3-D CTS:* To evaluate the coupling effect of adjacent TSVs in 3-D CTS, we implement the TSV-to-TSV coupling model presented in Section II and TSV-optimized arrangement method presented in Section III-D into our proposed flow. After exhaustively sweeping the TSV bound from 1 to 50, as shown in Fig. 15, considering the coupling effect of adjacent TSVs can further improve the skew and power consumption. Specifically, the improvement on the skew and power is more significant with the increase of TSV bound, while the area and number of whitespace blocks are kept unchanged. This phenomenon happens because more TSVs in the limited whitespace would

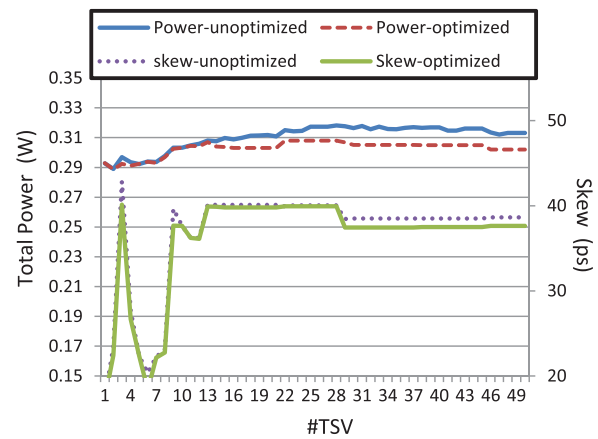


Fig. 15. Skew and power trends for ispd09f11 for different TSV bounds [1, 50] with or without optimizing TSV-to-TSV coupling effect.

aggravate the coupling effect of adjacent TSVs, if TSVs are not optimally arranged.

To evaluate the parasitic impact of TSVs on timing, we extract a last level tree from the whole 3-D clock network implemented with a real industry benchmark, which consists of one pair of sink nodes and a driving buffer as shown in Fig. 16. The wire length from the sink node to the parent node is $3.5 \mu\text{m}$. The load capacitance for the sink node is 0.538 fF . The experimental result shows that the parasitic effect of a single TSV can induce about 20 ps latency variation (from 7 to 27 ps). Note that for the whole 3-D clock network, the latency from the clock source to the clock sink is about 400 ps, whereas the skew is only 10 ps. Therefore, neglecting the parasitic effect of TSVs may lead to severe timing degradation, especially for the paths with more TSVs.

6) *Verification With Real Industry Benchmarks:* To further verify our 3-D CTS method, we also implement the proposed method with two real industry cases, one with 739 clock sinks and the other with 11447 clock sinks. Both of them are modules in AMD GPU processors. The distribution and information of all clock sinks are extracted from the original 2-D IC design. Then, we partition them into two layers and mark the available whitespace blocks for clock TSV

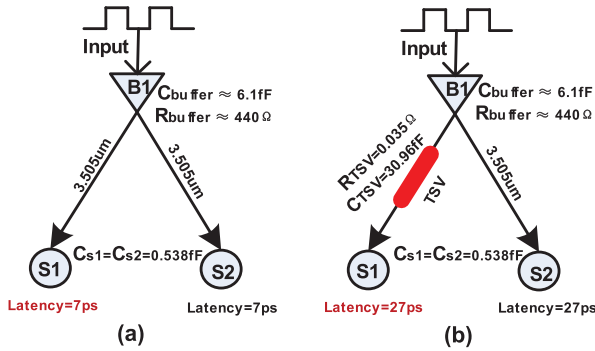


Fig. 16. Parasitic effect of TSV-induced latency. (a) The path latency without TSV. (b) The path latency with one TSV inserted in the left edge.

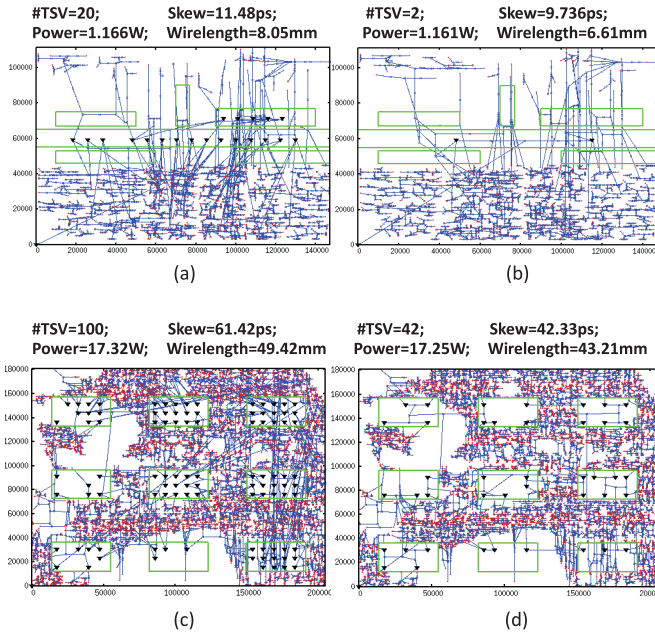


Fig. 17. Red points and black triangles: sinks and TSVs, respectively. Green rectangles: whitespace blocks. (a) With 739 clock sinks, traditional 3-D-MMM-DBM solution with TSV moving, which induces longer wirelength. (b) With 739 clock sinks, our proposed TSV whitespace-aware 3-D CTS solution. (c) With 11447 clock sinks, the traditional 3-D-MMM-DBM solution with TSV moving. (d) With 11447 clock sinks, our proposed whitespace-aware 3-D CTS solution.

insertion according to the floorplan as shown in Fig. 17. With these industry benchmarks, we compare our proposed TSV whitespace-aware 3-D CTS method with the traditional 3-D-MMM-DBM method. First, for these two cases, we set the TSV bound as 20 and 100, respectively. According to the results shown in Fig. 17(a) and (c), the traditional 3-D-MMM-DBM solution tends to use as many TSVs as the given TSV bound permits and leads to many longer wires due to moving TSVs into the limited whitespace blocks. In contrast, as Fig. 17(b) and (d) shows, our proposed solution uses only 2 and 42 TSVs, respectively, and can achieve better wirelength, skew, and power consumption. In addition, for the first case with 739 clock sinks, we explore the impact of TSV bound by sweeping the TSV bound from 1 to 50. The results in Fig. 18 show that for both the traditional 3-D-MMM-DBM and our methods, with the increase of TSV bound, the skew

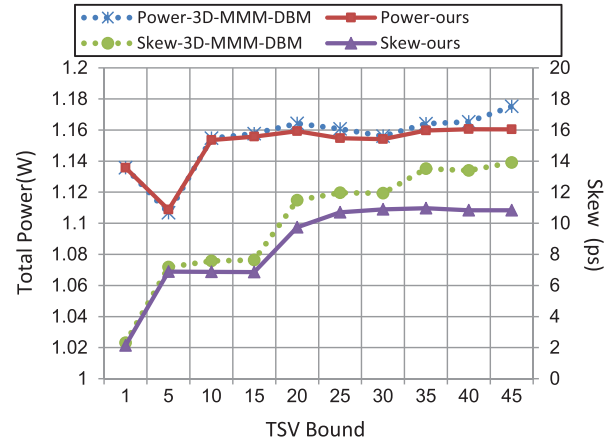


Fig. 18. Skew and power trends for a real industry benchmark based on different TSV bounds [1, 50].

and power consumption tend to be aggravated when TSV bound is larger than 15, that is, because of the excessive long wires induced by moving TSVs into the limited whitespace areas for the traditional 3-D-MMM-DBM method, and extra wire-snaking overhead when reconstructing merging segment in our proposed method. However, our proposed method still shows much more superiority than the traditional 3-D-MMM-DBM method with the increase of TSV bound.

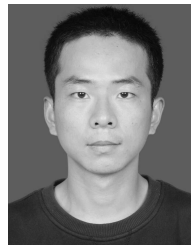
V. CONCLUSION

In this paper, we formulate the whitespace-aware TSV arrangement problem in 3-D CTS and propose a practical and efficient algorithm to solve this problem. The algorithm consists of three stages: sink preclustering, TWA-3-D-MMM topology generation, and DME merging segment reconstruction. By leveraging the TSV-to-TSV coupling model, we also propose an efficient clock TSV arrangement method to alleviate the coupling effect of adjacent TSVs. Experiment results show that our method is more practical and efficient, compared with the traditional 3-D-MMM method with TSV moving adjustment.

REFERENCES

- [1] Y. Xie, G. H. Loh, B. Black, and K. Bernstein, "Design space exploration for 3D architectures," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 2, no. 2, pp. 65–103, 2006.
- [2] M. Pathak, Y.-J. Lee, T. Moon, and S. K. Lim, "Through-silicon-via management during 3D physical design: When to add and how many?" in *Proc. ICCAD*, 2010, pp. 387–394.
- [3] M.-K. Hsu, Y.-W. Chang, and V. Balabanov, "TSV-aware analytical placement for 3D IC designs," in *Proc. 48th ACM/EDAC/IEEE DAC*, Jun. 2011, pp. 664–669.
- [4] T.-Y. Kim and T. Kim, "Bounded skew clock routing for 3D stacked IC designs: Enabling trade-offs between power and clock skew," in *Proc. IGCC*, Aug. 2010, pp. 525–532.
- [5] X. Zhao and S. K. Lim, "Power and slew-aware clock network design for through-silicon-via (TSV) based 3D ICs," in *Proc. 15th ASPDAC*, Jan. 2010, pp. 175–180.
- [6] T.-Y. Kim and T. Kim, "Clock tree synthesis for TSV-based 3D IC designs," *ACM Trans. Design Autom. Electron. Syst.*, vol. 16, no. 4, 2011, Art. ID 48.
- [7] X. Zhao, J. Minz, and S. K. Lim, "Low-power and reliable clock network design for through-silicon via (TSV) based 3D ICs," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 1, no. 2, pp. 247–259, Feb. 2011.

- [8] X. Zhao, D. L. Lewis, H.-H. S. Lee, and S. K. Lim, "Pre-bond testable low-power clock tree design for 3D stacked ICs," in *IEEE/ACM ICCAD Dig. Tech. Papers*, Nov. 2009, pp. 184–190.
- [9] X. Zhao, D. L. Lewis, H.-H. S. Lee, and S. K. Lim, "Low-power clock tree design for pre-bond testing of 3-D stacked ICs," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 5, pp. 732–745, May 2011.
- [10] M. Mondal *et al.*, "Thermally robust clocking schemes for 3D integrated circuits," in *Proc. DATE*, Apr. 2007, pp. 1–6.
- [11] J.-S. Yang, J. S. Pak, X. Zhao, S. K. Lim, and D. Z. Pan, "Robust clock tree synthesis with timing yield optimization for 3D-ICs," in *Proc. 16th ASPDAC*, Jan. 2011, pp. 621–626.
- [12] Y. Shang, C. Zhang, H. Yu, C. S. Tan, X. Zhao, and S. K. Lim, "Thermal-reliable 3D clock-tree synthesis considering nonlinear electrical-thermal-coupled TSV model," in *Proc. 18th ASP-DAC*, Jan. 2013, pp. 693–698.
- [13] M. P. D. Sai, H. Yu, Y. Shang, C. S. Tan, and S. K. Lim, "Reliable 3-D clock-tree synthesis considering nonlinear capacitive TSV model with electrical-thermal-mechanical coupling," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 11, pp. 1734–1747, Nov. 2013.
- [14] T.-Y. Kim and T. Kim, "Clock tree synthesis with pre-bond testability for 3D stacked IC designs," in *Proc. 47th ACM/IEEE DAC*, Jun. 2010, pp. 723–728.
- [15] T.-Y. Kim and T. Kim, "Resource allocation and design techniques of prebond testable 3-D clock tree," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 1, pp. 138–151, Jan. 2013.
- [16] S.-J. Wang, C.-H. Lin, and K. S.-M. Li, "Synthesis of 3D clock tree with pre-bond testability," in *Proc. IEEE ISCAS*, May 2013, pp. 2654–2657.
- [17] J. Minz, X. Zhao, and S. K. Lim, "Buffered clock tree synthesis for 3D ICs under thermal variations," in *Proc. ASPDAC*, Mar. 2008, pp. 504–509.
- [18] T.-Y. Kim and T. Kim, "Clock tree embedding for 3D ICs," in *Proc. 15th ASPDAC*, Jan. 2010, pp. 486–491.
- [19] X. Zhao and S. K. Lim, "Through-silicon-via-induced obstacle-aware clock tree synthesis for 3D ICs," in *Proc. 17th ASPDAC*, Jan./Feb. 2012, pp. 347–352.
- [20] X. Li, W. Liu, H. Du, Y. Wang, Y. Ma, and H. Yang, "Whitespace-aware TSV arrangement in 3D clock tree synthesis," in *Proc. IEEE Comput. Soc. Annu. ISVLSI*, Aug. 2013, pp. 115–120.
- [21] J. Kim, J. Cho, and J. Kim, "TSV modeling and noise coupling in 3D IC," in *Proc. 3rd ESTC*, Sep. 2010, pp. 1–6.
- [22] K. Yoon *et al.*, "Modeling and analysis of coupling between TSVs, metal, and RDL interconnects in TSV-based 3D IC with silicon interposer," in *Proc. 11th EPTC*, Dec. 2009, pp. 702–706.
- [23] C. Liu, T. Song, J. Cho, J. Kim, J. Kim, and S. K. Lim, "Full-chip TSV-to-TSV coupling analysis and optimization in 3D IC," in *Proc. 48th ACM/EDAC/IEEE DAC*, Jun. 2011, pp. 783–788.
- [24] T. Song *et al.*, "Analysis of TSV-to-TSV coupling with high-impedance termination in 3D ICs," in *Proc. 12th ISQED*, Mar. 2011, pp. 1–7.
- [25] H. Chaabouni *et al.*, "Investigation on TSV impact on 65 nm CMOS devices and circuits," in *Proc. IEEE IEDM*, Dec. 2010, pp. 35.1.1–35.1.4.
- [26] Y. Peng, T. Song, D. Petranovic, and S. K. Lim, "On accurate full-chip extraction and optimization of TSV-to-TSV coupling elements in 3D ICs," in *Proc. IEEE/ACM ICCAD*, Nov. 2013, pp. 281–288.
- [27] J. Cho *et al.*, "Active circuit to through silicon via (TSV) noise coupling," in *Proc. IEEE 18th Conf. EPEPS*, Oct. 2009, pp. 97–100.
- [28] M.-C. Tsai, T.-C. Wang, and T. 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. 2011.
- [29] M. A. B. Jackson, A. Srinivasan, and E. S. Kuh, "Clock routing for high-performance ICs," in *Proc. 27th ACM/IEEE DAC*, Jun. 1990, pp. 573–579.
- [30] T.-H. Chao, Y.-C. Hsu, J.-M. Ho, and A. B. Kahng, "Zero skew clock routing with minimum wirelength," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 39, no. 11, pp. 799–814, Nov. 1992.
- [31] F.-W. Chen and T. T. Hwang, "Clock tree synthesis with methodology of re-use in 3D IC," in *Proc. 49th ACM/EDAC/IEEE DAC*, Jun. 2012, pp. 1094–1099.
- [32] *ISPD'09 Benchmarks*. [Online]. Available: <http://ispd.cc/contests/09/ispd09cts.html>
- [33] *Predictive Technology Model*. [Online]. Available: <http://ptm.asu.edu/>
- [34] *NGSPICE*. [Online]. Available: <http://ngspice.sourceforge.net/>



Wulong Liu (S'13) received the B.S. degree from Microelectronic School, Xidian University, Xi'an, China, in 2010. He is currently pursuing the Ph.D. degree with the Department of Electronic Engineering, Tsinghua University, Beijing, China.

His current research interests include design automation, low-power design, 3-D ICs, VLSI design, optical interconnect, and 2.5-D/3-D system-on-a-chip integration. He has authored several papers in TVLSI, the *ACM Journal on Emerging Technologies in Computing Systems*, *IEEE Design & Test Magazine*, the Asia and South Pacific Design Automation Conference, the International Symposium on Quality Electronic Design, and the IEEE Computer Society Annual Symposium on VLSI.



Yu Wang (S'05–M'07–SM'14) received the B.S. and Ph.D. (Hons.) degrees from Tsinghua University, Beijing, China, in 2002 and 2007, respectively.

He is currently an Associate Professor with the Department of Electronic Engineering, Tsinghua University. His current research interests include parallel circuit analysis, application-specific hardware computing (in particular, the brain-related problems), and power and reliability-aware system design methodology. He has authored or co-authored over 100 papers in refereed journals and conferences.

Dr. Wang serves as an Associate Editor of the *IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN* and the *Journal of Circuits, Systems, and Computers*. He was the TPC Co-Chair of the 2011 International Conference on Field-Programmable Technology, is the Finance Chair of the 2012–2015 International Symposium on Low Power Electronics and Design (ISLPED), and serves as a TPC Member in many important conferences, including the Design Automation Conference, FPGA, DATE, the Asia and South Pacific Design Automation Conference (ASPDAC), the International Symposium on Low Power Electronics and Design, the International Symposium on Quality Electronic Design, the International Conference on Field Programmable Technology, and ISVLSI. He was a recipient of the IBM X10 Faculty Award in 2010, the Best Paper Award in ISVLSI 2012, the Best Poster Award in HEART 2012, and six Best Paper Nominations in ASPDAC/CODES/ISLPED.



Guoqing Chen (M'14) received the B.S. and M.S. degrees in electronic engineering from Tsinghua University, Beijing, China, in 1998 and 2001, respectively, and the Ph.D. degree in electrical engineering from the University of Rochester, Rochester, NY, USA, in 2007.

He was with Intel, Folsom, CA, USA, from 2007 to 2012, where he was involved in the physical design of integrated graphics in CPUs. After that, he joined AMD, Beijing, as a member of the technical staff, where he was involved in the clock and power delivery networks of discrete GPUs. He is currently with the AMD Research China Laboratory, Beijing. He has authored over 20 peer-reviewed journal and conference papers. His current research interests include low-power circuits and architectures, clock and power distribution networks, power and thermal modeling/management of multicore systems, and 3-D integrated circuits.



Yuchun Ma (M'06) received the B.S. degree in computer science from X'ian Jiaotong University, Xi'an, China, in 1999, and the Ph.D. degree in computer science from Tsinghua University, Beijing, China, in 2004.

She is currently an Associate Professor with the Department of Computer Science and Technology, Tsinghua University. She has authored over 100 papers in refereed journals and conferences. Her current research interests include physical design automation algorithm for ASIC and FPGA designs, optimization methodologies for 3-D ICs, and high-level synthesis algorithms.

Prof. Ma serves as the TPC Chair of the 2014 International Conference on Field-Programmable Technology (ICFPT), and has served as the ASPDAC TPC Member since 2010. She is a Steering Committee Member of ASPDAC and the Finance Chair of ICFPT 2010.



Yuan Xie (M'97) received the B.S. degree in electronic engineering from Tsinghua University, Beijing, China, and the M.S. and Ph.D. degrees in electrical engineering from Princeton University, Princeton, NJ, USA.

He is currently a Professor with the Department of Electrical and Computer Engineering, University of California at Santa Barbara, Santa Barbara, CA, USA. His current research interests include computer architecture, design automation, VLSI design, and embedded system.

Prof. Xie has served as the TPC Chair of ASPDAC 2013 and TPC Vice Chair of ASPDAC 2012. He also served as the General Co-Chair and TPC Co-Chair of ISLPED 2014 and 2013, respectively. He is currently an Associate Editor of the *ACM Journal of Emerging Technologies in Computing Systems*, the *IEEE TRANSACTIONS ON VERY LARGE SCALE INTEGRATION SYSTEMS*, the *IEEE TRANSACTIONS ON COMPUTER AIDED DESIGN OF INTEGRATED CIRCUITS*, the *IEEE Design & Test Magazine*, and *IET Computers and Digital Techniques*.



Huazhong Yang (M'97–SM'00) was born in Ziyang, China, in 1967. He received the B.S. degree in microelectronics and the M.S. and Ph.D. degrees in electronic engineering from Tsinghua University, Beijing, China, in 1989, 1993, and 1998, respectively.

He is currently a Professor with the Department of Electronic Engineering, Tsinghua University. He is also a Professor of Yangtze River Scholars authorized by the Ministry of Education. His current research interests include the internet-of-things chip

design and related application systems, system-on-a-chip low-power circuits and systems, and EDA technology.

Prof. Yang served as an Associate Editor of the *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS-II* from 2010 to 2013. He is also an Associate Editor of the *International Journal of Electronics* and the *Journal of Circuits, Systems, and Computers*.