Clock Tree Synthesis for Heterogeneous 3-D Integrated Circuits

Isuru Daulagala and Ioannis Savidis

Department of Electrical and Computer Engineering
Drexel University, Philadelphia, PA 19104
idaulagala@nvidia.com, isavidis@coe.drexel.edu

***Abstract*—Heterogeneous integration of disparate device planes is a benefit of through-silicon-via based 3-D integrated circuits (ICs). Clock delivery for heterogeneous 3-D ICs requires novel circuit techniques and algorithms as compared to 2-D and even homogeneous 3-D ICs. Novel algorithms for topology generation (heterogeneous 3-D balanced bipartitioning) and for embedding of a clock tree in a heterogeneous 3-D stack (heterogeneous 3-D deferred merge and embedding) are described. The algorithms are applied to a two-tiered heterogeneous stack of dies with configurations using the 28 nm, 40 nm, and 65 nm technology nodes. The results are analyzed against existing process agnostic clock tree synthesis algorithms for 3-D ICs, showing a reduction of 7% in power consumption, 6% in delay, 4% in buffer count and 4% in wirelength.**

I. Introduction

Through-silicon-via (TSV) based 3-D ICs provide a means to increase transistor density beyond scaling. In addition, 3-D integration addresses the limitations of traditional 2-D ICs due to increasing global interconnect wirelength, which impacts circuit area, power, and delay. Furthermore, TSV based 3-D ICs allow for heterogeneous integration in a 3-D stack. Dies from disparate process nodes and with different functions such as logic, RF, and memory are vertically integrated to form a stacked system, as shown in Figure 1. The general use of TSV based 3-D ICs requires: 1) physical design tools that account for the third dimension [1], 2) methods to address the thermal constraints of the stacked circuits, and 3) improvements in the yield of the stacked die.

Clock tree synthesis (CTS) is a primary step in physical design. Restricting the global skew of a clock distribution network (CDN) is a primary objective since it directly impacts the performance of the circuit. CDNs consume a large portion of the dynamic power of a circuit. Minimizing the power consumption of a CDN is, therefore, an important design objective. CTS is composed of three primary steps: 1) topology generation, 2) embedding or clock tree routing, and 3) buffer insertion. Modern CTS algorithms for 2-D ICs are primarily based on the deferred merge and embedding (DME) algorithm [3]–[7]. The DME takes a topology as an input and embeds (finds locations for internal nodes) the clock tree in two phases: 1) a bottom up phase where the zero skew merging segments of the parent of each pair of nodes are determined, and 2) a top down phase where the exact location for the child node is determined based on the minimum wirelength from the parent to the child. The loci of points that are a specific Manhattan distance away from each child, described as a Tilted Rectangular Region (TRR), are generated to construct a merging segment in a Manhattan plane. The intersection of two TRRs are chosen as the merging segment that can either be a Manhattan arc (a line segment with slope 1 or -1) or a point. The majority of the current CTS algorithms use the Elmore delay [8] to calculate the zero skew merging segments.

Existing literature on CTS for 3-D ICs and CDN Design [9]–[14] address homogeneous dies in the 3-D stack. Pavlidis et. al [13], [14] considered different topologies in a fabricated 3-D IC. Minz et. al [9] proposed a thermal aware CTS algorithm called BURITO that accounts for the increased thermal density of 3-D ICs. BURITO expanded the Method of Means and Medians (MMM) proposed by Jackson et. al [15] to 3-D ICs, generating a topology for the 3-D clock tree. In [10], Kim and Kim proposed an optimal layer assignment algorithm for internal nodes of a topology, which is described as Deferred Layer Embedding (DLE-3D). In [11], the Nearest Neighbor algorithm proposed by Edahiro [16] is used to generate a topology that results in a CDN with less TSVs, reduced wirelength, and reduced delay than the one generated

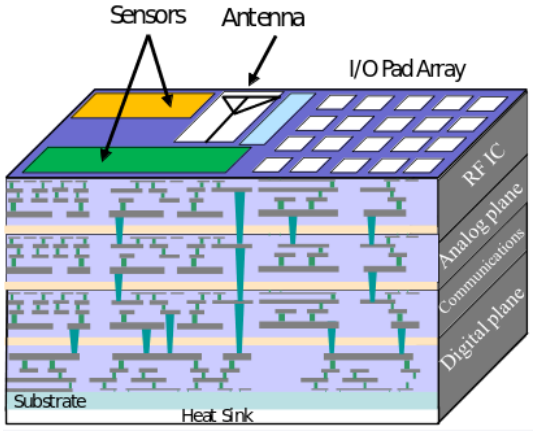


Fig. 1: A heterogeneous 3-D IC that includes dies from differing process technologies [2].

by the 3D-MMM algorithm. All of the described 3-D CTS algorithms are based on homogeneous dies using the 45nm Predictive Technology Model (PTM) [17].

The contributions of this work are summarized as follows:

- The first to propose CTS algorithms for heterogeneous dies with disparate process nodes. Two algorithms are proposed:
  1) Heterogeneous 3-D balanced bipartitioning (H3D-BB) for topology generation, and
  2) Heterogeneous 3-D deferred merge and embedding (H3D-DME) for embedding (physical placement of internal nodes) of a given topology
- The proposed algorithm is the first to consider the TSV location when embedding a node in a heterogeneous 3-D stack during CTS. In the existing literature, the location of a TSV between a parent and a child was ignored since the parasitic impedances of both dies are equivalent.

II. Problem formulation

The problem of heterogeneous clock tree synthesis is defined as follows:

Problem - Heterogeneous clock tree synthesis: *Given 1) a set of sinks $S = \{s_1, s_2, ..., s_n\} \subset \mathbb{R}^3$, 2) a source s_0 in a heterogeneous 3-D stack with per unit length wire resistance and capacitance values of, respectively, r_{w,l_i} and c_{w,l_i} for each die i, and 3) TSV resistance and capacitance values of, respectively, r_{TSV} and c_{TSV}, construct an abstract tree topology $G(S)$ and an embedded clock tree $T(G(S))$ that minimizes the cost function*

$$Cost = \sum_{\forall l_i}(\alpha_{l_i} \cdot \sum_{\forall x_{l_i}} |e_{x_{l_i}}|) + \beta \cdot nv(T_x) + \sum_{\forall l_i} \gamma_{l_i} \cdot buffers_{l_i}, \quad (1)$$

while satisfying the zero skew constraint, where $nv(T_x)$ is the number of TSVs in a binary tree with root node x and $|e_{x_{l_i}}|$ is the edge length from x to the parent node. The cost due to the total number of buffers on each tier, l_i, is given by the third term in (1). The cost function given by (1) is based on work described in [10]. The α, β, and γ parameters are weights associated with, respectively, the relative cost of a wire, TSV, and a buffer. Both the wire weight α_{l_i} and buffer weight γ_{l_i} are determined for each device tier l_i.

***Subproblem* - Construction of heterogeneous clock tree topology:** *Given 1) a set of sinks $S = \{s_1, s_2, ..., s_n\} \subset \mathbb{R}^3$, 2) a source s_0 in a heterogeneous 3-D stack with per unit length wire resistance r_{w,l_i} and capacitance c_{w,l_i} for each tier l_i, and 3) TSV resistance r_{TSV} and capacitance c_{TSV}, construct an abstract tree topology $G(S)$.*

The topology generation algorithm must consider the impedance mismatches between dies. In a topology, all node locations except that of the leafs (sinks) are unknown. Therefore, when trying to find the parent of two non-leaf nodes, it is difficult to determine the layer each node belongs to. The problem of layer assignment is further exasperated in a heterogeneous stack since the layer position of each node impacts heuristics that are used to determine the tree topology.

***Subproblem* - Heterogeneous clock tree topology embedding:** *Given 1) a set of sinks $S = \{s_1, s_2, ..., s_n\} \subset \mathbb{R}^3$, 2) a source s_0 in a heterogeneous 3-D stack with interconnect per unit length resistance r_{w,l_i} and capacitance c_{w,l_i} for each tier l_i, 3) TSV resistance r_{TSV} and capacitance c_{TSV}, and 4) an input abstract tree topology $G(S)$, embed the internal nodes of the topology such that the power and number of TSVs of the stack are minimized while meeting the zero skew constraint.*

When embedding a node in a heterogeneous die stack, the layer position significantly impacts the clock tree. As will be discussed, there are multiple zero skew locations for a parent. In a homogeneous stack, moving a zero skew location to a separate tier potentially increases the delay from the parent to each child. However, in a heterogeneous die stack, selecting a zero skew location on another tier can potentially reduce the delay from the parent to each child.

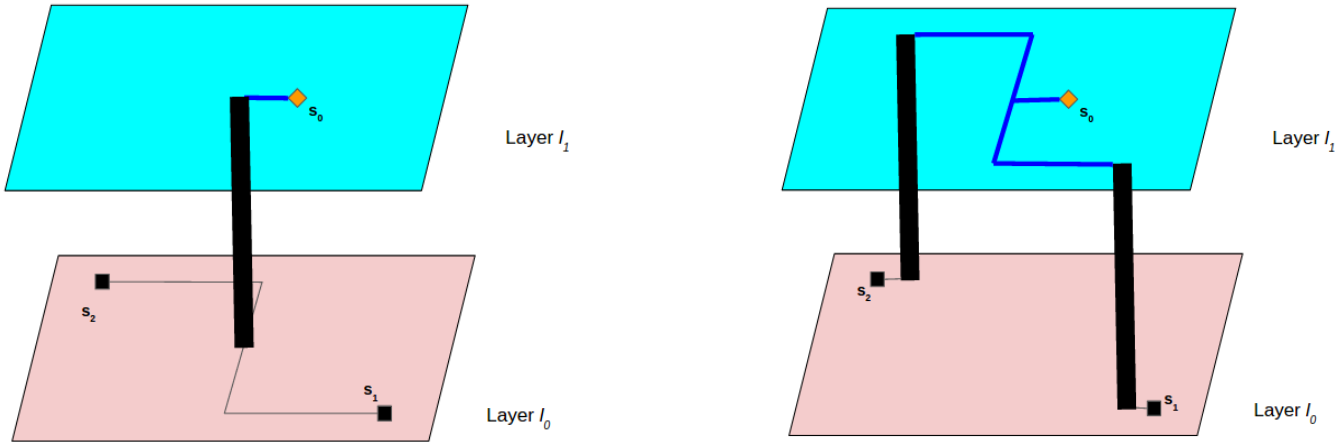
III. Heterogeneous 3-D Clock Tree Synthesis

The primary objective of heterogeneous 3-D CTS is to determine the optimal route of the clock tree while accounting for the impedance mismatches between the device planes. Two potential interplane routing configurations are shown in Figure 2. Node s_0 is the source on die l_1, which has a significantly lower impedance as compared to die l_0, on which sinks s_1 and s_2 are located. In Figure 2(a), the majority of the clock tree is routed on the high impedance tier l_0 with one TSV used to route the signal from l_1 to l_0. In Figure 2(b), however, the majority of the clock is routed on the low impedance tier l_1 and two TSVs route the signal from l_0 to l_1. Although the source to sink wirelengths are similar, as shown in Figure 2(b), the delay from source to each sink is shorter. The decrease in delay, however, requires an additional TSV, which potentially leads to an increase in power consumption as compared to a single TSV between a parent and the two children. The decision to move a node to the low impedance tier depends on if the move decreases the overall capacitance of the circuit.

A. Heterogeneous 3-D Topology Generation

In order to account for impedance mismatches between tiers, a binary tree topology is created that favors routing on the lower impedance tiers. Determining the nodes that share a parent during the execution of the topology generation algorithm becomes more complex in a heterogeneous stack. In [11], the tier of the parent node is tentatively set to the intermediate tier of the children, which is permissible as the impedance characteristics of each tier are equal. However, with impedance mismatches on each tier, the device plane on which a node is placed significantly impacts the heuristic used to generate a topology. A novel topology generation algorithm, Heterogeneous 3-D Balanced Bipartitioning (H3D-BB), is therefore proposed to account for the impedance mismatches as well as the delay through a TSV. H3D-BB is based on the 2-D Balanced Bipartitioning(BB) algorithm from [6].

A visual representation of the topology generation for both 2-D and 3-D ICs is shown in Figure 3. The traditional



(a) (b)

Fig. 2: Multiple ways to route a clock signal in 3-D stack with heterogeneous dies. Clock tree generated with a majority of the path routed on layer (a) l_0 and (b) l_1.

2-D BB algorithm is shown in Figure 3(a). In H3D-BB, an octagonal bounding box is generated similar to the 2-D algorithm, projecting all sinks from all device planes onto a single x,y plane. The sinks on tier 0 are depicted in blue and black, while the sinks in tier 1 are depicted in red and orange. The sinks that contribute to the octagonal boundary are depicted in red and black. The BB algorithm is modified for heterogeneous ICs (H3D-BB), where the H3D-BB algorithm accounts for impedance mismatches between layers and the routing cost through TSVs. A non-uniform octagonal prism is used to represent the topology generated by the H3D-BB algorithm, as shown in Figure 3(c). The octagonal bases of the prism are scaled up or down to represent the impedance mismatches between layers. The height of the prism is equal to the capacitance through the TSV multiplied by a constant k where $k > 1$. The constant k represents the weight of the capacitance of a TSV with respect to that of an interconnect. A k equal to 1 corresponds to the clock tree with the largest number of TSVs. The differences between H3D-BB and the 2-D implementation in [6] include the weight w_r and the distance metric d for two nodes. When two sinks are on two different dies, calculating d accounts for the TSV capacitance, capacitances of each tier, and the 2-D Manhattan distance x between the two nodes p and r, as given by (2). When the two sinks are on the same device tier, only the 2-D Manhattan distance x and the capacitance of the given tier are used for the calculation of the distance metric.

$$d(p,r) = (C_{l_p} \cdot x + C_{l_r} \cdot x)/2 + k \cdot C_{TSV} \tag{2}$$

$$d(p,r) = (C_{l_p} \cdot x + C_{l_r} \cdot x)/2 \tag{3}$$

B. Heterogenous 3-D Embedding

The DLE-3D algorithm described in [11] is used to assign temporary tiers for internal nodes of the topology. In order to exploit the differences in impedance between tiers, the bottom up phase of DME-3D [11] is modified, as described by Algorithm 1.

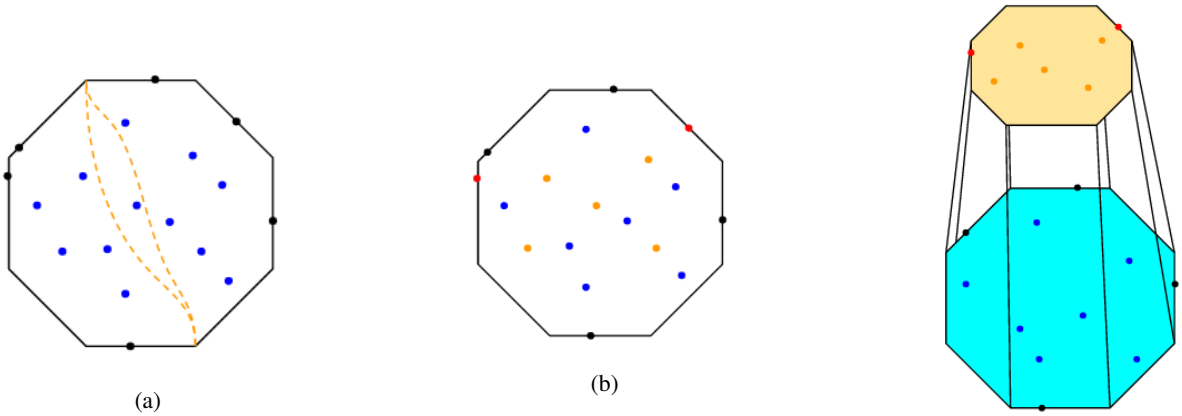
Execution of the algorithm that moves a node to a low impedance layer is shown in Figures 4 and 5. When both children are on the same device plane, there are multiple zero skew points. If the children are on a higher impedance plane as compared to the other device planes, the node x, which is the parent of the children, is a candidate to move to a low impedance plane. The node is moved depending on two factors:

1) Is the parent of x (the grandparent of the children to merge) on the same tier as the children? If yes, then moving the node to the low impedance plane costs two TSVs. If not, the node is a candidate for moving.
2) Are the two children far enough apart such that the benefits of a move to the low impedance tier outweigh the increase in delay and power due to the TSV? If the Manhattan distance between the children exceeds a value g, the node is a candidate to move to a lower impedance tier.

If both factors are met, the node is moved to the tier with lower impedance, as shown in Figure 4.

When the children are on two different device planes, there are multiple zero skew points. The one on the lower impedance device plane has a lower delay from the parent to the children. The lower impedance plane is chosen if the grandparent is also on the same plane.

The modifications to the bottom up portion of the 3D-DME algorithm are described by Algorithm 1. Merging segments of nodes are moved to low impedance tiers as previously described. The top down portion of DME-3D is applied without any change in H3D-DME. The implementation of H3D-BB for topology generation and H3D-DME for embedding is described as H3D-CTS.



(a) (b) (c)

Fig. 3: H3D-BB algorithm. The partitioning of nodes into sink subsets is shown in (a) for the 2-D BB algorithm [6]. The sink sets are recursively partitioned based on the minimum partition cost, which depends on the sum of the Manhattan diameters of each partition. The initial octagon with sinks from different device planes used in H3D-BB is shown in (b). The sinks from each tier are denoted by different colors. The two device planes are separated and scaled to construct a non-uniform octagonal prism as shown in (c).

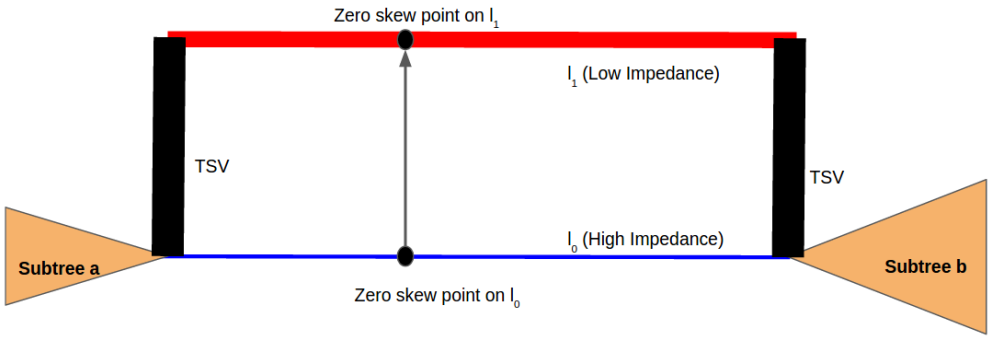


Fig. 4: Moving a parent node to a low impedance layer if both children are on a higher impedance layer. Since there are multiple zero skew points, the node is moved if additional TSVs are not inserted or the distance between the children is large enough that the impact of an additional TSV on power and delay is compensated by the gains from routing on a lower impedance plane.

IV. Simulation Results

A. Simulation Setup

The developed algorithms were implemented in C++14/STL and were run on a Linux server with a 3.0 GHz Intel Xeon E5-2690 processor and 64 GB of memory. The technology parameters for the wires and buffers used are listed in Table I. In addition to the technology parameters, a TSV with a resistance r_{TSV} of 0.05 Ω and a capacitance c_{TSV} of 15 fF was also included in the simulations.

An evaluation of the algorithms was completed using the benchmark circuits from the 2009 ISPD clock network synthesis contest [18]. Since the benchmarks were composed of 2-D circuits, a conversion to a 3-D stack with two dies was performed by scaling the dimensions by a factor of $1/\sqrt{2}$, positioning sinks in each die randomly, and removing layout obstacles that block certain areas of the die from clock tree placement. Six circuit configurations were evaluated by using the first three ISPD 2009 circuits and assigning each die to different process nodes. The 3-D stack containing 40 nm and 65 nm dies for the benchmark circuits includes the suffix *a*, and the stack with 28 nm and 40 nm dies with the suffix *b*. SPICE simulations were completed to determine the delay (skew) and quantify power when V_{DD} was set to 1.1 V and

TABLE I: Interconnect and buffer technology parameters used for simulation.

Parameter		Technology 28nm	40nm	65nm
Wire	$r_w(\Omega/\mu m)$	0.21	0.12	0.04
	$c_w(fF/\mu m)$	0.29	0.22	0.11
Buffer	$t_d(ps)$	12	16	22
	$r_b(\Omega)$	64	70	94
	$C_{in}(fF)$	6	9	10

Algorithm 1 Bottom up algorithm to build a tree of segments in DME

procedure BUILD TREE OF SEGMENTS($G(S), C_{TSV}, C_{l_i}$)
  for all Nodes $n \in G(S)$ in bottom-up order **do**
    if n is a sink **then**
      $ms(n) = PL(V)$ ▷ ms= Merging Segment, PL= Physical Location
    else
      $(a,b) = CHILDREN(n)$
      if a and b on the same layer **then**
        if If a and b are on the lower impedance layer **then**
          Find $ms(n)$ in the same layer as a and b
        else
          if $d(a,b) > g$ and if the parent of n is lower impedance layer **then**
            Find $ms(n)$ in lower impedance layer
          end if
        end if
      else
        if Parent of n is on lower impedance layer **then**
          Find $ms(n)$ on lower impedance layer
        else
          Find $ms(n)$ on the same layer as parent of n
        end if
      end if
    end if
  end for
end procedure

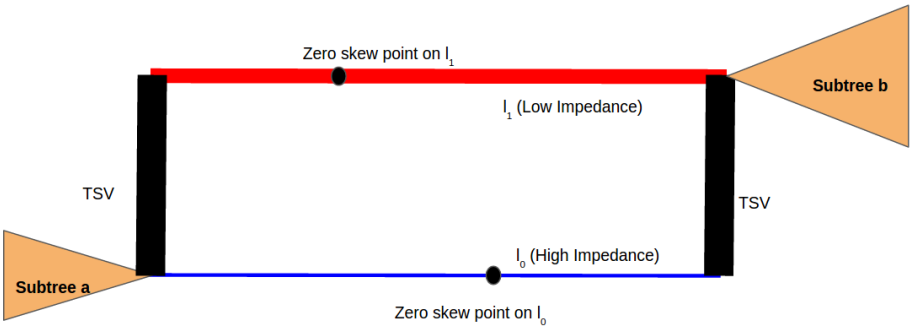


Fig. 5: Choosing a correct tier for a parent when the children are on different tiers. There are two zero skew points. The zero skew point in the low impedance plane is chosen if the grandparent of the merging nodes is also on the low impedance tier.

the frequency to 1 GHz. Since it is not possible to directly extend the topology generation algorithm NN-3D [11] to a heterogeneous stack, the proposed flow is compared with that of a flow comprised of 3D-MMM [9] for topology generation and both DLE-3D and DME-3D from [11] for embedding. The latter flow is labeled as 'base' in Tables II and III. Results from a comparison of TSV count and wirelength are listed in Table II. The bounded number of TSVs for 3D-MMM is set to 100%. The value of k is set to 1 in (2), which corresponds to the highest TSV bound.

In order to account for slew, buffers were inserted when the loading capacitance of the wires exceeded 300 fF, similar to [11]. The delay, skew, power, and buffer count are compared with the base flow in Table III.

TABLE II: Comparison of the number of TSVs and wirelength between the base (3D-MMM) and H3D-CTS algorithms.

Benchmark	**#TSVs**		**Wirelength**	
	Base	H3D-DME	Base	H3D-DME
ispd09f11a	46	47	134480	127494
ispd09f11b	46	46	134480	127494
ispd09f12a	45	45	126627	123834
ispd09f12b	45	44	126630	123834
ispd09f21a	42	44	139258	134587
ispd09f21b	42	43	139258	134587
Ratio	**1.00**	**1.01**	**1.00**	**0.96**

B. Discussion of Results

The results listed in Table II indicate that the H3D-CTS algorithm outperforms the combination of the 3D-MMM and

TABLE III: Comparison of delay, power, skew, and buffer count between the base (3D-MMM and 3D-DME) and H3D-CTS algorithm.

Benchmark	Delay		Skew		Power		Buffers	
	Base	**H3D-DME**	**Base**	**H3D-DME**	**Base**	**H3D-DME**	**Base**	**H3D-DME**
ispd09f11a	0.36	0.33	25	26	58.64	53.23	158	149
ispd09f11b	0.44	0.42	26	26	61.78	58.66	161	151
ispd09f12a	0.35	0.33	24	25	55.56	51.59	159	152
ispd09f12b	0.41	0.39	25	25	58.68	54.45	161	156
ispd09f21a	0.38	0.36	26	27	61.27	57.34	164	165
ispd09f21b	0.46	0.43	26	27	67.78	65.44	165	166
Ratio	**1.00**	**0.94**	**1.00**	**1.03**	**1.00**	**0.93**	**1.00**	**0.96**

3D-DME (base) algorithms in wirelength while the TSV counts remain approximately the same. The improved reduction in wirelength of H3D-CTS is attributed to the efficacy of the H3D-BB topology generation algorithm as compared to the 3D-MMM algorithm. The optimizations for stacked heterogeneous dies lead to an increase in the number of TSVs as compared to the base algorithm since nodes are assigned to lower impedance tiers, which requires routing between planes. By assigning nodes to low-impedance tiers, the total power consumption of the 3-D IC is reduced, as indicated by the results listed in Table III. On average, the power consumption is reduced by 7%, the number of buffers is reduced by 4%, and the source to sink delay is decreased by 6%, while increasing the total number of TSVs by a negligible margin (~1%). The average clock skew increased by 3%, with a maximum skew of 27 ps, below the set skew bound of 30 ps. The overall wirelength is reduced by 4%, which is primarily due to the proposed topology generation algorithm H3D-BB outperforming the 3D-MMM algorithm. Although the proposed algorithms and simulations were executed for a two die 3-D stack, both can be extended to a 3-D stack with any number of tiers.

V. Conclusion

Current CTS algorithms for 3-D ICs do not account for heterogeneous device planes. CTS algorithms for a 3-D stack with heterogeneous device planes are required to fully benefit from TSV based 3-D ICs. Two key algorithms for synthesizing clock trees in heterogeneous TSV-based ICs have been proposed: 1) Heterogeneous 3-D balanced bipartitioning (H3D-BB), which is a fast and effective (in terms of wirelength, power, and delay) top down topology generation algorithm, and 2) heterogeneous 3-D deferred merging and embedding (H3D-DME), which embeds internal nodes of a topology. The H3D-DME algorithm includes key optimizations made to the DME-3D algorithm proposed in [11] for heterogeneous device planes. Through simulation results, it has been shown that the proposed algorithms produce cost effective clock trees for 3-D ICs with heterogeneous device planes as compared to existing algorithms which do not account for the heterogeneity in a die stack. The proposed algorithms produce clock trees that consume 6% less power, reduce delay by 4%, and require 4% fewer buffers.

References

[1] S. K. Lim, "Tsv-aware 3d physical design tool needs for faster mainstream acceptance of 3d ics," *ACM DAC Knowledge Center (dac.com)*, pp. 321–326, 2010.

[2] V. F. Pavlidis and E. G. Friedman, *Three-dimensional integrated circuit design*. Morgan Kaufmann, 2010.

[3] M. Edahiro, "Minimum skew and minimum path length routing in vlsi layout design," *NEC research & development*, vol. 32, no. 4, pp. 569–575, Oct. 1991.

[4] T.-H. Chao, J.-M. Ho, and Y.-C. Hsu, "Zero skew clock net routing," in *Proceedings of the ACM/IEEE Design Automation Conference*, Jun. 1992, pp. 518–523.

[5] K. D. Boese and A. B. Kahng, "Zero-skew clock routing trees with minimum wirelength," in *Proceedings of Fifth Annual IEEE International ASIC Conference and Exhibit*, Nov. 1992, pp. 17–21.

[6] T.-H. Chao, Y.-C. Hsu, J.-M. Ho, and A. Kahng, "Zero skew clock routing with minimum wirelength," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 39, no. 11, pp. 799–814, Nov. 1992.

[7] J. Cong and C.-K. Koh, "Minimum-cost bounded-skew clock routing," in *Proceedings of the IEEE International Symposium on Circuits and Systems*, May 1995, pp. 215–218.

[8] J. Rubinstein, P. Penfield Jr, and M. A. Horowitz, "Signal delay in rc tree networks," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 2, no. 3, pp. 202–211, Jul. 1983.

[9] J. Minz, X. Zhao, and S. K. Lim, "Buffered clock tree synthesis for 3d ics under thermal variations," in *Proceedings of the 2008 Asia and South Pacific Design Automation Conference*, Mar. 2008, pp. 504–509.

[10] T.-Y. Kim and T. Kim, "Clock tree embedding for 3d ics," in *Proceedings of the Asia and South Pacific Design Automation Conference (ASP-DAC)*, Jan. 2010, pp. 486–491.

[11] ——, "Clock tree synthesis for tsv-based 3d ic designs," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 16, no. 4, pp. 48:1–48:21, Oct. 2011.

[12] X. Zhao, J. Minz, and S. K. Lim, "Low-power and reliable clock network design for through-silicon via (tsv) based 3d ics," *IEEE Transactions on Components, Packaging and Manufacturing Technology*, vol. 1, no. 2, pp. 247–259, Dec. 2011.

[13] V. F. Pavlidis, I. Savidis, and E. G. Friedman, "Clock distribution networks for 3-d integrated circuits," in *Proceedings of the IEEE Custom Integrated Circuits Conference*, Sep. 2008, pp. 651–654.

[14] ——, "Clock distribution networks in 3-d integrated systems," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 12, pp. 2256–2266, Dec. 2011.

[15] M. A. Jackson, A. Srinivasan, and E. S. Kuh, "Clock routing for high-performance ics," in *Proceedings of the ACM/IEEE Design Automation Conference*, Jun. 1990, pp. 573–579.

[16] M. Edahiro, "A clustering-based optimization algorithm in zero-skew routings," in *Proceedings of the ACM/IEEE Design Automation Conference*, Jun. 1993, pp. 612–616.

[17] Y. Cao, T. Sato, D. Sylvester, M. Orshansky, and C. Hu, "Predictive technology model," *Internet: http://ptm. asu. edu*, 2002.

[18] C. N. Sze, P. Restle, G.-J. Nam, and C. Alpert, "Ispd 2009 clock network synthesis contest," in *Proceedings of the 2009 International Symposium on Physical Design*, Apr. 2009, pp. 149–150.

[19] I. Savidis and E. G. Friedman, "Closed-form expressions of 3-d via resistance, inductance, and capacitance," *IEEE Transactions on Electron Devices*, vol. 56, no. 9, pp. 1873–1881, Sep. 2009.