INTEGRATION the VLSI journal (xxxx) xxxx–xxxx

Contents lists available at ScienceDirect

INTEGRATION, the VLSI journal

journal homepage: www.elsevier.com/locate/vlsi

Revisiting 3DIC benefit with multiple tiers

Wei-Ting Jonas Chan[a], Andrew B. Kahng[a,b], Jiajia Li[a,*]

[a] ECE Departments, University of California, San Diego, United States
[b] CSE Departments, University of California, San Diego, United States

ARTICLE INFO

Keywords:
3D integration
Power and area benefits
Infinite dimension

ABSTRACT

3DICs with multiple tiers are expected to achieve large benefits (e.g., in terms of power, area) as compared to conventional planar designs. However, few if any previous works study *upper bounds* on power and area benefits from 3DIC integration with multiple tiers. In this work, we use the concept of *implementation with infinite dimension* to estimate upper bounds on power and area benefits achievable by 3DICs versus 2DICs. We observe that the maximum power benefit even with infinite dimension can be only 18% versus 2DIC for particular designs. Such benefits reduce further under assumptions of inter-tier variation. We confirm our observation by performing 3D benefit estimation across various technologies. Our study also indicates that it is typically difficult for pure logic-logic 3D integration to achieve a simultaneous (10%, 10%, 10%) improvement in (performance, power, area/cost) compared to the conventional 2D implementation. In addition, we study power of designs across various dimensions (e.g., pseudo-1D, 2D, 3D with two, three and four tiers).[1] We observe that design power sensitivity to implementation with different dimensions correlates well with placement-based Rent parameter of the netlist. Therefore, placement-based Rent parameter can possibly be a simple indicator of 3D power benefit. Our study also shows that netlist synthesis and optimization should be aware of the target implementation dimension (e.g., 2D versus 3D). Finally, we use a simple example to show that there remain potential large 3DIC benefits versus 2DIC for (block-based) SoC designs.

1. Introduction

Three-dimensional integrated circuits (3DIC) with multiple tiers is a promising technology in the "More-than-Moore" era to integrate more functionality with greater bandwidth and less power. Many previous works propose 3DIC optimization approaches to achieve better design quality over conventional planar implementations. Further, due to higher integration and reduced wirelengths, 3DICs with more than two tiers are expected to offer larger benefits (e.g., less power). A recent work [39] shows that 3DICs with three tiers achieve 15% more power reduction as compared to corresponding two-tier 3DIC implementations and 36% power reduction versus 2D implementations. A much smaller body of work addresses the fundamental question of predicting 3DIC benefits over conventional 2D implementation, and upper-bounding these benefits. Chan et al. in [4] derive a 67% upper bound of wirelength reduction from a two-tier 3DIC over 2D designs. However, no previous works propose *upper bounds* on the power and total cell area reductions achievable by 3DICs over 2D designs.[2]

In this work, we revisit the 3DIC benefit in terms of power and area with multiple tiers. More specifically, we propose the concept of *implementation in infinite dimension* (that is, where all gates can be placed as close as possible – essentially, adjacent – to each other) to derive an upper bound on 3D power and area benefits for given design, technology, and tool/flow. Such implementation in infinite dimension is achieved by synthesis and netlist optimization with zero wireload model (0-WLM).[3] Our studies show that the power benefits can only be 18% for particular designs, even with an infinite-dimensional layout resource. Moreover, we study the maximum potential (performance, power, area/cost) benefits of 3DICs. None of our testcases is able to achieve more than (10%, 10%, 10%) improvement from 3D integration, a somewhat disappointing observation relative to the hoped-for benefits from 3DIC. We further evaluate design power across various dimensions (i.e., pseudo-1D, 2D, 3D with multiple tiers). We observe that design power sensitivity to different implementation dimensions correlates with Rent parameters of netlists, especially placement-based

* Corresponding author.
E-mail addresses: wechan@ucsd.edu (W.-T.J. Chan), abk@cs.ucsd.edu (A.B. Kahng), jil150@ucsd.edu (J. Li).

[1] In this paper, the pseudo-1D implementation indicates the design implementation with high aspect ratio layout.
[2] In our following discussion, we refer to the power and total cell area reductions as well as frequency improvement achievable by 3DICs over 2D designs as 3D benefits.
[3] We note that the upper bound on 3D benefits is specific to the given optimization tool/flow. However, since we use a leading-edge commercial P & R tool in our experiments, we believe our estimated upper bound is not far from the true upper bound.

http://dx.doi.org/10.1016/j.vlsi.2017.01.004
Received 16 September 2016; Accepted 12 January 2017

Rent parameters. Based on this observation, we suggest that netlist synthesis should be aware of implementation dimension so as to minimize design power[1].

We summarize our contributions as follows.

- We propose the concept of implementation with *infinite dimension* (i.e., netlist optimization with 0-WLM), based on which, we study the upper bound on power and area benefits of 3DICs.
- We show that upper bounds on 3DIC power and area benefits can be quite small – at most 39% power reduction and 10% area reduction even with infinite dimension.
- We perform 3D benefit estimation across various technologies and compare the 3D benefits versus the benefits from an improved P & R tool/flow in 2D implementation.
- We study the maximum potential (performance, power, area/cost) benefits of 3DICs. Our results indicate that it is typically difficult to achieve a simultaneous (10%, 10%, 10%) improvement of (performance, power, area/cost) from pure logic-logic 3D integration.
- We study design power sensitivity to various implementation dimensions (i.e., pseudo-1D, 2D, 3D with different tier numbers and infinite dimension) and show the empirical evidence of a correlation between the power sensitivity and the Rent parameter of the netlist.
- We suggest that there is potential benefit from netlist synthesis optimizations being aware of the implementation dimension.
- Using a simple example with macros and/or blockages, we show that 3D benefits can be large for (block-based) SoC designs.

The remainder of this paper is organized as follows. Section 2 reviews related works on 3DIC optimization and prediction of 3DIC benefits. Section 3 describes our implementation flows as well as design power and area estimation flows in different dimensions. In Section 4, we describe experimental setup and results that quantify 3DIC power benefits and power sensitivity to dimensions. Section 5 concludes and gives directions for ongoing work.

2. Related works

Various approaches have been proposed for implementation and optimization of 3DICs. Table 1 summarizes area, power and wirelength benefits reported in previous works. In the table, "—" indicates "not applicable", i.e., not addressed in the cited work. We note that although we show total cell area reported by previous works in the table, area reduction is typically not the major objective in 3DIC optimization. In addition, most of these works use wirelength reduction as their major design objective [2,6–9,13,14,18,32,33].

We first review previous works for integration and optimization of 3DICs. Liu et al. [21] propose transistor-level 3D monolithic integration. Bobba et al. [3] propose a cell-mapping and placement flow for monolithic 3D. Thorolfsson et al. [41] propose a 3D placer based on mPL. Lim [20] obtains power benefit by studying the capacitance reduction in TSV-based 3D implementations. Song et al. [39] propose a block-level folding approach and show corresponding power benefits of a three-tier stacking. Chang et al. [5] apply the flow [31] to 7 nm technology node. Nayak et al. [27], Athikulwongse et al. [1], and Panth et al. [30] study the power benefits from various 3D integrations (e.g., monolithic 3D, mini-TSV, and TSV-based integration). Song et al. [38] study power reduction with consideration of the power distribution network. Jung et al. [11,12], Lee et al. [19], and Ok et al. [28] achieve power benefits by applying block-level integration to the *OpenSparcT2* processor, a multicore GPU, and a stereo image processor.

Estimation of 3D power benefits has also drawn much attention. Priyadarshi et al. [34] propose an architectural framework to estimate 3DIC power for design space exploration. Kim et al. [15–17] propose models to estimate wirelength and power reductions based on buffer insertion. Chan et al. [4] propose a machine learning-based methodology to estimate power benefits of (3DIC) design flows, and apply their methodology to the flow proposed by Panth et al. [31]. However, no existing work studies potential *upper bounds* on 3DIC power and area benefits. To our knowledge, ours is the first work to address this gap.

Table 1
Summary of 3D benefits studied in previous works.

Work	% Benefit (area)	% Benefit (power)	% Benefit (WL)	# tier
[1]	7%	9%	WL increases	2
[2]	—	—	50%	2
[3]	—	15%	13%	2
[4]	—	39%	—	2
[5]	—	17%	46%	2
[6]	—	—	54%	4
[7]	—	—	50%	2
[8]	—	—	30%	2
[9]	—	—	26%	3
[11]	8%	21%	26%	2
[12]	—	20%	9%	2
[13]	—	—	20%	2
[14]	—	—	4%	2
[15]	—	—	37%	2
[16]	—	28%	41%	4
[17]	—	23%	28%	4
[18]	—	—	32%	2
[19]	—	22%	—	2
[20]	—	3%	25%	2
[21]	—	7%	16%	2
[24]	—	37%	48%	2
[27]	—	37%	—	2
[28]	—	13%	14%	2
[30]	—	35%	33%	2
[32]	—	—	19%	2
[33]	—	—	40%	2
[34]	—	20%	—	2
[38]	—	19%	—	2
[39]	3%	36%	42%	3
[41]	—	13%	21%	2

In this work, we also examine the correlations between netlist properties and 3D power benefits. Our studies suggest that netlist synthesis could benefit from awareness of implementation dimension. Previous works which study the relationship between netlist structures and placement and routing (P & R) quality of results (QoR) include those of Liu and Marek-Sadowska [22,23], which propose metrics to predict 2D P & R wirelength from netlist structure. They also propose optimization during the technology mapping stage of logic synthesis to improve 2D P & R results. Rahman et al. [36] propose a low-power gate-sizing scheme using a rich library with complex and large-size cells for logic synthesis, and a library with only simple cells for P & R. Seo et al. [37] argue that the benefit of using complex cells in advanced nodes will diminish due to routing congestion. However, [37] does not consider how 3D integration might mitigate the routing congestion seen in a 2D design implementation. (We consider this aspect of 3D integration below.)

3. Implementation in various dimensions

We now describe our implementation and benefit estimation flows across various dimensions – pseudo-1D, 2D, 3D with multiple tiers, and infinite dimension.

3.1. Pseudo-1D implementation

To estimate the power penalty of design implementations in a limited dimension, we propose *pseudo-1D implementation*, that is, design implementation with high aspect ratio layout. In this work, we refer to an implementation with layout aspect ratio 0.1 (block height equal to block width /10) as a pseudo-1D implementation.

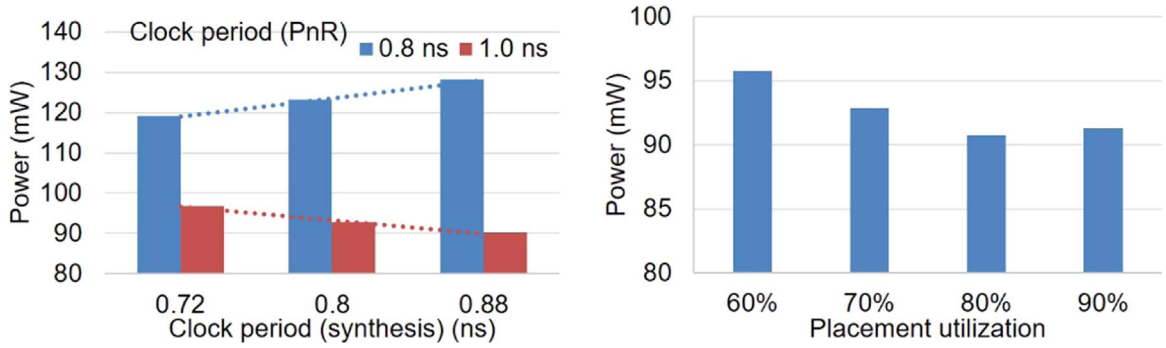

Fig. 1. Design power with (i) varying synthesis clock period, and (ii) placement utilization. Design: JPEG. Technology: 28FDSOI.

3.2. Optimal 2D implementation

We pursue an "optimal" 2D implementation so as to quantify the true benefits from 3D integration with multiple tiers and from implementation in infinite dimension. To achieve this, we obtain multiple conventional planar implementations by sweeping several key parameters such as synthesis clock period, placement utilization and BEOL stack options; we then select the best (e.g., minimum power) outcome. Fig. 1 shows an example where we vary the synthesis clock period and placement utilization for different 2D implementations. We observe that when the design has tight timing constraints, slightly smaller synthesis clock period eventually leads to smaller power after placement and routing. On the other hand, for a design with loose timing constraints, slightly larger synthesis clock period results in smaller design power after routing.

Furthermore, design power versus placement utilization exhibits a roughly unimodal behavior. In other words, if a placement is too compact, the power increases due to routing congestion (detouring, crosstalk, etc.). On the other hand, if the placement is too sparse, the power again increases due to longer wirelength (larger wire capacitance). Regarding BEOL stack options, we vary the number of layers from six to 11 in our experiments and compare the resultant design power after routing. However, we observe that the power variation is quite small (e.g., less than 3% for design JPEG). This is because the six-layer stack is able to offer enough routing resources for the designs and technology used in our experiments. We therefore implement our design testcases by varying (i) synthesis clock period (e.g., 0.9x, 1.0x and 1.1x of clock period used in P & R) and (ii) placement utilization (e.g., 60%. 70%, 80%, 90%), then selecting the outcome with minimum power consumption.[4]

We also apply such implementation flows to other dimensions to obtain (as far as we are able) fair comparisons.

3.3. Power benefit estimation for 3DICs

Given that there is no "golden" 3DIC implementation flow, we propose an implementation flow, based on the conventional 2D implementation tools, to achieve an optimistic estimation of design quality of 3DICs with multiple tiers. With the Shrunk2D flow proposed in [31] as a starting point, we perform the flow described in Algorithm 1 to estimate 3DIC benefits with multiple tiers.[5] To estimate the reduction of wire parasitics from shrunk footprint area, we shrink the cell LEF by a factor of $1/\sqrt{T}$ in both height and width, where T is the number of tiers. We also apply the same scaling ratio to shrink the BEOL LEF to ensure that routing resources remain adequate. We implement designs based on the shrunk LEF (Line 1). With the shrunk LEF implementations, we then divide the die area into M x M grids. For each grid, we perform iterative min-cut partitioning to divide the cells within the grid into T clusters which are assumed to be placed on T tiers (Line 3). Details of our partitioning procedure are described in Algorithm 2. In the procedure, we iteratively apply min-cut partitioning based on Fiduccia-Mattheyses (FM) algorithm to partition the cells into two parts with area ratio of $(T - k)$ where k is the iteration index. The min-cut bipartitioning is performed with MLPart [46]. After each partitioning, the cells from the smaller-area part will be assigned to a new tier. We then collapse the cells in the smaller-area part into one super cell, fix the super cell in the first partition, set its area to zero and continue with the partitioning procedure. The iterative partitioning optimization terminates when all cells are assigned to a tier. Based on the partitioning solution, we annotate parasitics to nets which have cells from different tiers. More specifically, we calculate the maximum delta in tier depth across all cells connected to the net, and for each unit of (delta in tier) depth, we annotate RC corresponding to one TSV and vias across six metal layers (Line 4). With the annotated RC, we perform incremental sizing, VT-swapping and buffer insertion optimization to fix timing violations.

Note that in our estimation flow, we can estimate the benefits from wire parasitic reduction in 3DICs. In addition, we ignore the potential performance and power penalties from placement legalization (when we allocate cells to different tiers), from routing congestion caused by cross-tier power delivery, from difficulties in clock tree synthesis in a 3DIC, and from additional die-to-die variability margin. Our proposed flow therefore provides an optimistic estimation of 3DIC design qualities.[6][7] In our study, we also attempt to back-annotate placement solution with shrunk LEF to physical synthesis optimization. However, our experimental results show that a such back-annotation loop does not lead to further power benefits in the routed design.[8] We therefore apply a one-pass flow in our experiments.

[4] Our separate studies perform implementations with fine-grained choices of synthesis clock period (e.g., 0.8x to 1.2x with a step size of 0.05x of the P & R clock period) and placement utilization (e.g., 50–90% with a step size of 5%). Results for the JPEG testcase show that the optimal solution (i.e., solution with minimum power) found with fine-grained parameter sweeping is only 2% better than the solution found with our reported methods. We also attempt to vary the synthesis utilization (i.e., the utilization of floorplan as input to physical synthesis). However, experimental results show that for the technology and design testcases studied, sweeping of synthesis utilization does not lead to any power benefits as compared to a flow without physical synthesis.

[5] We have Shrunk2D flow from [31]. Since the flow runs on an old version of P & R tool (i.e., *Cadence SoC Encounter vEDI10.1*) where the results are even worse than the 2D implementation with *Cadence Innovus Implementation System v15.2*, we do not use the Shrunk2D flow for 3D benefits estimation.

[6] Due to lack of production-quality 3D design tool/flow, we are unable to precisely capture the penalties from routing congestion, cross-tier power delivery networks, clock tree synthesis, etc.

[7] In our estimation flow we do not consider area impact of vertical interconnects (e.g., through silicon vias), as in [26]. This results in an optimistic estimation of 3D *area* benefits, with degree of optimism dependent on area overheads of vertical interconnects.

[8] Based on our experience, physical synthesis typically improves maximum performance when the clock constraints are tight. However, due to the pessimism of wireload in the physical synthesis, we rarely observe power reduction from physical synthesis optimization.

Algorithm 1. Evaluation of 3DIC benefit with T tiers.

1: Perform 2D implementation with shrunk LEF (scaling ratio=1/$\sqrt{T}$ in cell height and cell width)
2: Divide die area into M x M grids uniformly
3: Apply FM-based min-cut partitioning for T tiers.
4: Annotate RC according to tier assignment based on partitioning solution
5: Perform incremental optimization
6: Report design power

Algorithm 2. FM-based min-cut partitioning for T tiers.

Input: Netlist N, number of tiers T
Output: Subnetlists Sol={N_1, N_2,..., N_T}
1: $Sol \leftarrow \varnothing$
2: **for** k=1: $T-1$ **do**
3: $area_1 \leftarrow N.\ area/T$
4: $area_2 \leftarrow \frac{T-k}{T} \cdot N.\ area$
5: {N_k, N_{k+1}}=2WayMinCutPart(N, $area_1$, $area_2$) // tolerance=5%
6: $N \leftarrow$ collapse N_k into one super cell c_k
7: $c_k.\ area \leftarrow 0$
8: fix c_k in the first partition
9: $Sol \leftarrow Sol \cup \{N_k \setminus \{c_1, c_2, \ldots, c_k\}\}$
10: **end for**
11: **return** Sol

3.4. Implementation in infinite dimension

To estimate the upper bound on power and area benefits from 3DICs, we propose the concept of implementation with "infinite dimension", where we ignore wire parasitics during the implementation. To achieve this, we perform netlist optimization with zero wireload model (0-WLM).[9][10] Given that benefits from 3D integrations mainly come from the reduced wire parasitics in a shrunk footprint area, such implementation with infinite dimension is able to provide an upper bound on 3DIC benefits.

4. Experimental setup and results

We perform experiments in a 28 nm/12-track FDSOI foundry technology with dual-VT libraries, and 0.85 V nominal supply voltage. We perform experiments on an ARM CORTEXM0 core, three design blocks (AES, JPEG, VGA) from OpenCores [47], and LEON3MP from the ISPD-12 benchmark suite [29]. Parameters of these five testcases are shown in Table 2. For each design, we determine a range of clock periods starting from a clock period with relative loose timing constraint, up to the clock period which is close to the minimum clock period of the given design and technology. These designs are synthesized using *Synopsys Design Compiler vH-2013.12-SP3* [48] and then placed and routed using *Cadence Innovus Implementation System v15.2* [45]. We further use *Cadence Tempus Timing Signoff Solution v15.2* [49] for timing and power analysis, with wire parasitics (SPEF) obtained from Innovus.

Table 2
Benchmark parameters.

Design	#Instances	#Flip-flops	Clock period range
CORTEXM0	~9 k	840	0.8–1.0 ns
AES	~12 k	530	0.6–0.9 ns
JPEG	~43 k	4712	0.8–1.1 ns
VGA	~72 k	17057	0.7–1.0 ns
LEON3MP	~460 k	108817	1.1–1.3 ns

4.1. Evaluation of 3D benefits

We first compare design power and total cell area across various implementation dimensions across different clock periods. Fig. 2 shows the power and area comparison. All the implemented designs have no hold violation and a setup violation less than 10 ps. We observe that the maximum power benefits (i.e., the gap between the red curve versus the orange curve) from implementations in infinite dimension are respectively 36%, 39%, 20%, 18% and 26% for CORTEXM0, AES, JPEG, VGA and LEON3MP. The results show a large variation of 3D benefits across different designs. In addition, the power benefits from 3D integration with two, three and four tiers are less than 10% for designs JPEG, VGA and LEON3MP. Furthermore, we observe that the area benefits are small (i.e., <10% for all designs, and <4% for designs JPEG and VGA).

We further assess the upper bounds on potential PPAC (performance, power and area/cost) improvements from 3D integration based on our concept of infinite dimension.[11] Specifically, we implement designs in infinite dimension and sweep the clock period from a relatively large value (e.g., the maximum clock period shown in Fig. 2) to the minimum achievable clock period, with a step size of 50 ps. We also implement designs in infinite dimension at a higher supply voltage (i.e., 0.95 V) to explore the power-area tradeoff (i.e., more area benefits at the cost of larger power). We then compare the performance, power and area of the implementations in infinite dimension versus those of our "best possible" 2D implementations, and obtain tuples of potential maximum (performance, power, area/cost) benefits.

Fig. 3 shows the (performance, power, area/cost) tuples, where the results in the left column use low-frequency 2D implementations (i.e., clock periods={1.1 ns, 1.1 ns, 0.9 ns, 1.0 ns, 1.4 ns} for designs {CORTEXM0, JPEG, AES, VGA, LEON3MP}) as references; and the results in the right column use high-frequency 2D implementations (i.e., clock periods={0.8 ns, 0.8 ns, 0.6 ns, 0.7 ns, 1.1 ns} for designs {CORTEXM0, JPEG, AES, VGA, LEON3MP}) as references. Furthermore, we only show data points with all non-negative values in the tuple. Fig. 3 shows that for a single metric, the maximum improvement of performance, power or area can be ~40%, ~40% or ~10%, respectively, without degradation on the other two metrics. However, no design is able to achieve (10%, 10%, 10%) benefits from 3D integration, and only one design (i.e., CORTEXM0) shows more than (10%, 10%, 5%) benefits. Moreover, we observe that the area benefits from 3D integration are typically small (i.e., ≤5% for most of the data points); this matches the results in Fig. 2. These results may indicate a limited upside of pure logic-logic 3D integration.

4.2. A more realistic evaluation

As discussed in Section 3, our 3D power estimation ignores potential larger clock skew due to inter-tier process variation [43]. To achieve a more realistic estimation of 3D benefits, we quantify the

[9] To ensure a fair comparison to implementations at 2D and 3D, we perform netlist optimization with the same synthesis, placement and clock tree synthesis tool/flow but with 0-WLM and without any routing. Specifically, we use *Synopsys Design Compiler vH-2013.12-SP3*, [48] to synthesize the netlist, and use 0-WLM during the synthesis; we use *Cadence Innovus Implementation System v15.2*, [45] to perform placement and clock tree synthesis, and scale the interconnect RC by a very small number (i.e., 10^{-6}) during the placement and clock tree synthesis.

[10] Since our testcases are not hold-critical, the number of hold buffer insertions is negligible even when testcases are implemented with infinite dimension. To achieve an upper bound on 3DIC benefits, one might need to disable hold timing optimization during the implementation with infinite dimension.

[11] For "area/cost", we mean "area or cost". We assume that the cost is measured as area.

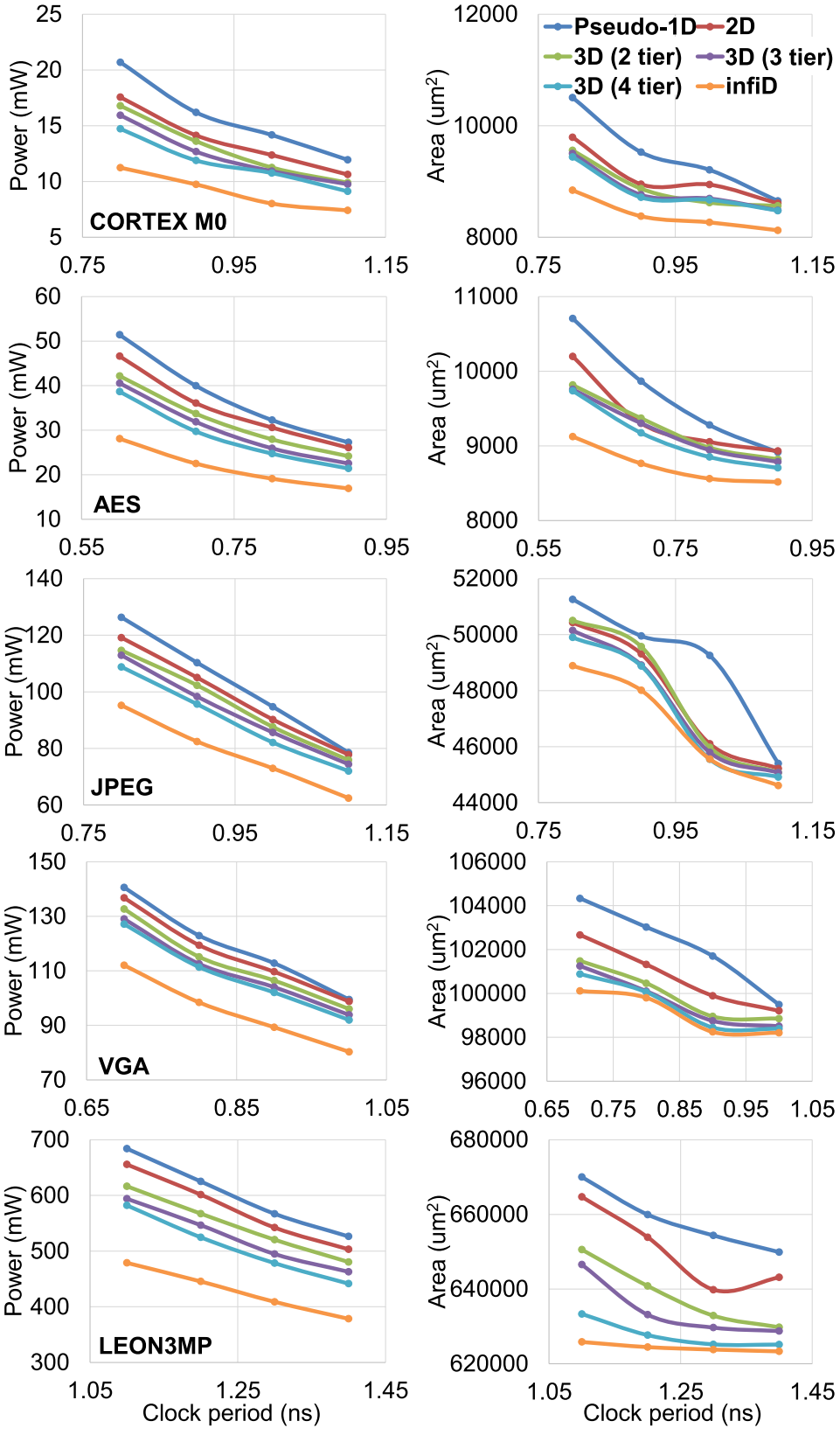

Fig. 2. Design power and total cell area evaluated across various implementation dimensions.

impact of clock skew on 3D power reduction. In our experiments, we enable multi-corner optimization by using both slow- and fast-corner libraries during the P & R stage. We further model potential clock skew increase due to difficulties in 3D clock tree synthesis (CTS) as well as inter-tier process variation by applying 0%, 5% and 10% clock uncertainties of the clock periods. The power benefits against the clock uncertainties are shown in Fig. 4. The results show that the 3D power benefits diminish when the clock uncertainties increase from 0% to 10% even for two designs which originally have the largest 3D benefits among our benchmarks. More specifically, the power benefit of a two-

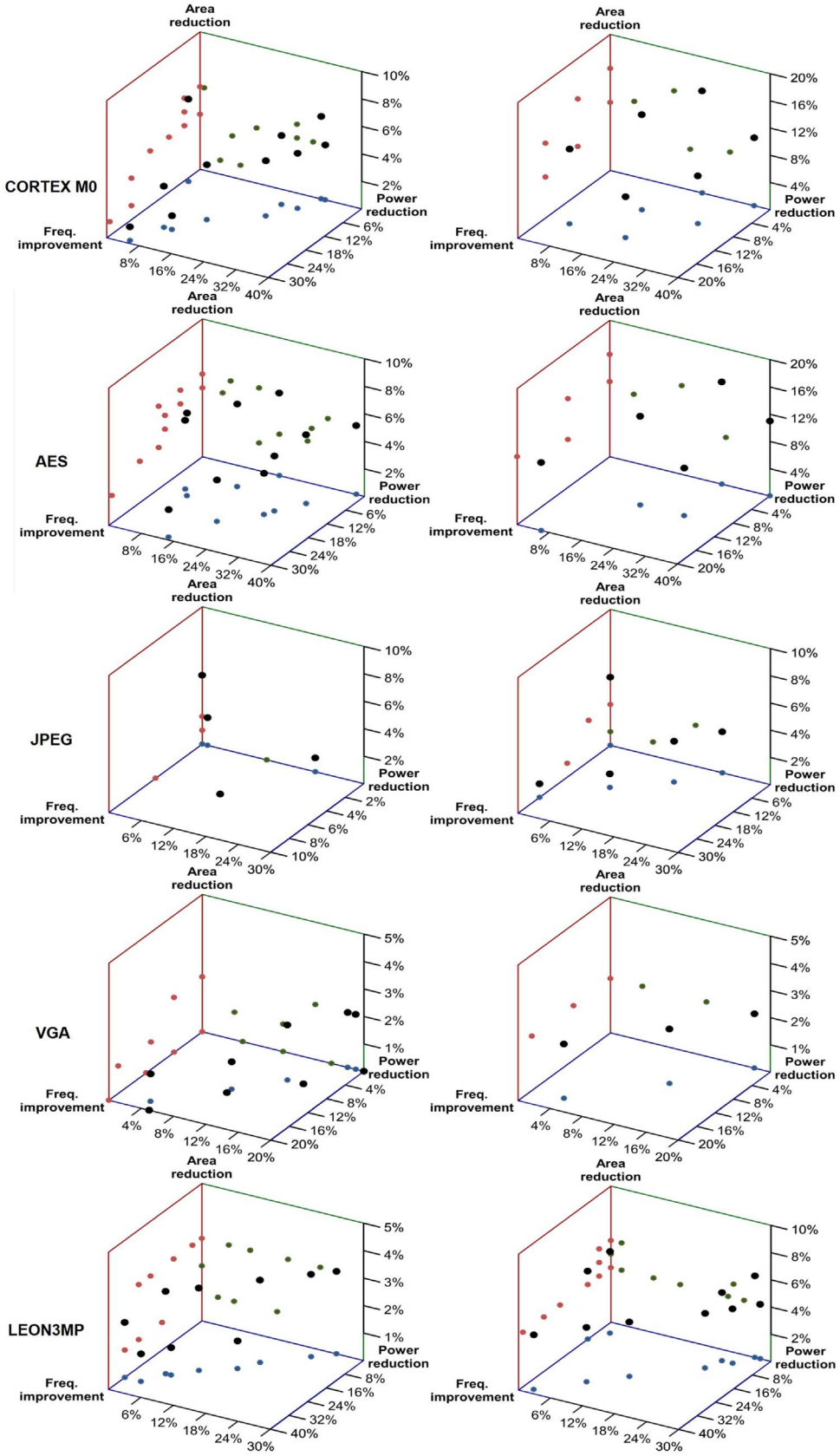

Fig. 3. Maximum potential (performance, power, area/cost) improvements from 3D integration. Left: Using low-frequency 2D implementations as references. Right: Using high-frequency 2D implementations as references. Blue, red and green dots are respectively projections on performance-power, performance-area and power-area planes.

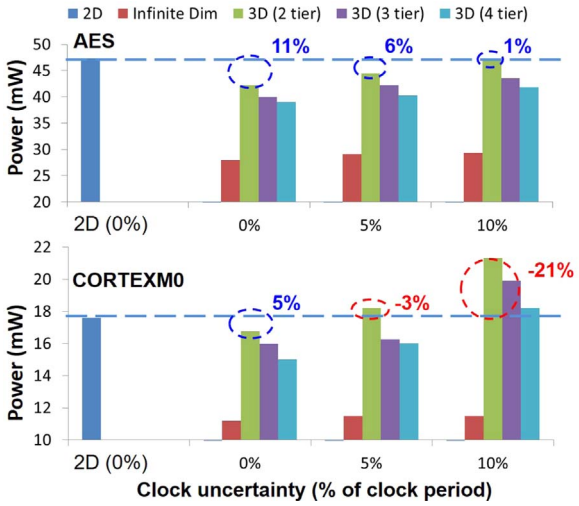

Fig. 4. The impact of larger clock skew (due to complexity of 3D CTS as well as inter-tier variation) on 3D power benefits.

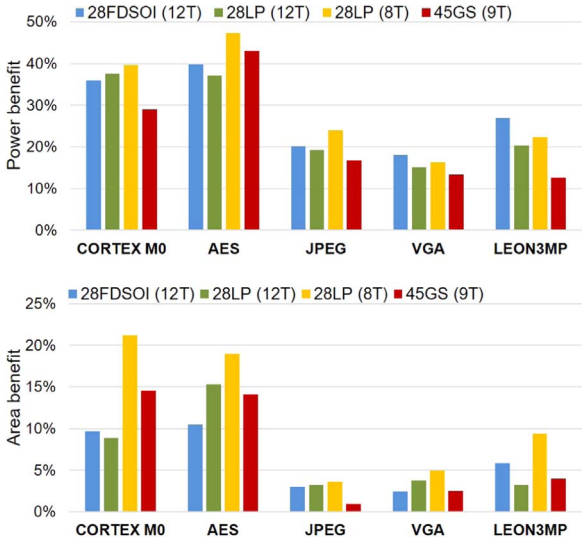

Fig. 6. 3D benefits assessed in infinite dimension showing consistency across various technologies.

tier 3D implementation decreases from 11% to 1% for AES and from 5% to −21% for CORTEXM0. Our observation indicates that it is critical for 3D clock tree optimization to minimize the impact of inter-tier variation on clock skew and latency.

4.3. Comparison between improved 2D versus 3D

In this subsection, we study the power and area benefits from an improved P & R tool/flow in the conventional 2D implementation, and compare these benefits versus the estimated 3D benefits. Fig. 5 shows the normalized power and area values of the 2D implementation using the latest version of a commercial P & R tool (i.e., *Cadence Innovus Implementation System v16.1*) (2D+) and the estimated 3D benefits with two tiers based on *Cadence Innovus Implementation System v15.2* (3D (2 tier)), with respect to 2D results using *Cadence Innovus Implementation System v15.2*. We observe similar power reductions from the improved 2D P & R tool/flow and 3D integration with two tiers. Quite interestingly, for particular designs (e.g., CORTEXM0 and LEON3MP), the power benefits from the improved P & R tool/flow are even higher than the estimated 3D benefits. The results may indicate that even one EDA company's year-to-year improvement is of similar magnitude to 3D benefits for particular designs. Fig. 5 further shows that the area benefits from both the improved P & R tool/flow and 3D integration are small in our results.

4.4. Assessment of 3D benefit across technologies

We further assess the upper bound on 3D benefits across different technologies. Fig. 6 shows the power and area benefits estimated in infinite dimension in 28FDSOI, 28LP (with 8 T and 12 T cells) and 45GS technologies. We use tight timing constraints with respect to each technology in our implementation. We observe consistent results across different technologies – (i) benefits on CORTEXM0 and AES are relatively higher than those on other designs, and (ii) area benefits are typically smaller than power benefits (especially on JPEG and VGA). Moreover, we observe larger benefits in technologies with weaker driving strength (i.e., 28LP with 8 T cells), where the wireload impact is larger on cell area and power.

4.5. Netlist study

We additionally study the possibility of correlations between (3D) power benefits and various netlist parameters (such as fanout distribution, slack distribution, sequential graph, Rent parameter, etc.) of designs. We observe that the power benefits are well correlated with Rent parameters.

We use the Rentian circuit generator *gnl* by Stroobandt et al. [40] to generate netlists with different Rent parameters, and we evaluate these netlists' power consumption across various implementation dimensions. The inputs to gnl are (i) number of cells, (ii) the target Rent parameter, (iii) the ratio between flip-flops and combinational cells, and (iv) the maximum path delay constraint. The gnl software starts with a set of standard cells and randomly inserts connections among the cells to form logic cones. The gnl software determines number of pins (of standard cells) and input/output terminals according to the user-specified Rent parameter. During the netlist generation, gnl recursively clusters logic cones to form larger ones. The number of terminals on the boundaries of the merged logic cones also follows the specified Rent parameter. The generated netlists thus have desired

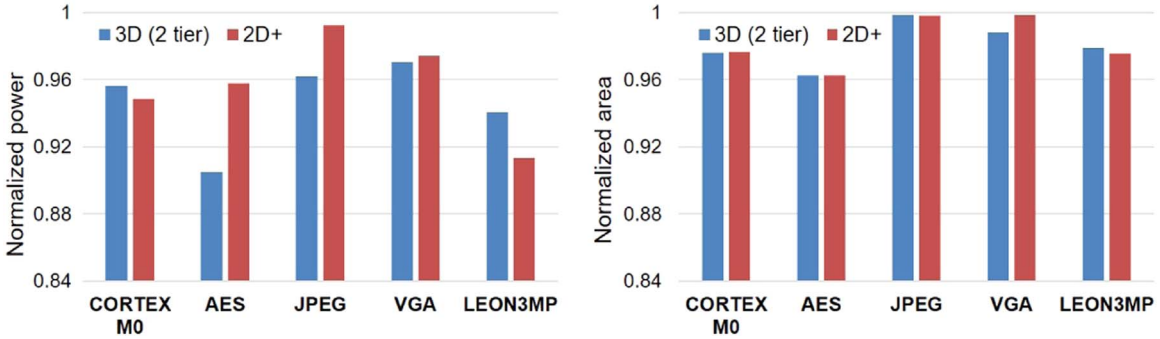

Fig. 5. Comparison between estimated 3D power and area benefits with two tiers versus power and area reductions from the latest version of a commercial P & R tool (i.e., 2D+).

Rent parameters by construction.[12] Table 3 summarizes our generated testcases. We generate netlists using cells from the foundry 28 nm 12-track FDSOI library and implement them using the flows described in Section 3.3. The initial generated netlists (with different Rent parameters) have similar power and area (i.e., within 3% difference) to help establish a fair comparison of power benefits across various Rent parameters. We define timing constraints such that the initial generated netlists have negative slacks, thus inducing non-trivial P & R optimizations.[13] Furthermore, to maintain a similar Rent parameter throughout the P & R flow (i.e., avoiding netlist restructuring), we apply a size-only restriction to all cell instances during the P & R optimization flow.

Fig. 7 shows the relationship between post-P & R power and netlist Rent parameter across various implementation dimensions. We observe that the power of the conventional 2D implementation increases with higher Rent parameter, whereas power increase (with Rent parameter) is smaller with 3D implementations. This suggests that implementations in higher dimension can mitigate power penalties due to higher-degree topologies of interconnections, which are indicated by larger Rent parameter values. Accordingly, more 3D power benefit may be expected with netlists having larger Rent parameter. We also observe the existence of *thresholds of Rent parameters* beyond which 3D power benefits seem to increase more rapidly (e.g., 0.69 in Fig. 7). Quantitative analysis of the relationship between power benefits and Rent parameter values will be one of our future works.

We also perform similar studies with realistic designs. Fig. 8 shows correlation between the maximum 3DIC power benefits estimated in infinite dimension, and Rent parameter values. We extract Rent parameters of the netlists using both partitioning-based and placement-based methods, where we assume that one pin (terminal) is induced by each cut hyperedge. Partitioning-based Rent parameter values are extracted based on recursive bipartitioning using the min-cut hypergraph partitioner MLPart [46]. To calculate placement-based parameter values, we perform fast placement with a commercial P & R tool [45] without any sizing, VT-swapping or buffering optimizations. We then perform rectangle sampling based on the placement solutions to estimate Rent parameters.

Even for a larger testcase (LEON3MP with 436 K instances), the runtime of the placement used to evaluate the placement-based Rent parameter is only 16 min. Our results show that the placement-based Rent parameter can possibly be a simple indicator of 3DIC power benefit for a given netlist.

In light of the correlation between power sensitivity to implementation dimensions and the netlist Rent parameter, we propose to modulate the cell usage in synthesis stage to control the Rent parameters and achievable 3D power benefit. We categorize library cells in the 28 nm FDSOI design enablement according to their input pin numbers – (1) one-input cells (buffers and inverters) (2) two-input cells (NAND2, NOR2, etc.) (3) three-input cells (NAND3, AOI21, etc.) (4) four-input cells (NAND4, OAI22, etc.), and (5) > four input cells (AOI212, MUX41, etc.). We then scale cell area in Liberty files to modulate the cell usage during synthesis so as to achieve netlists with different Rent parameters. More specifically, we choose design JPEG (which originally has a small Rent parameter) and scale down the area of complex cells; this induces the synthesis tool to use more complex cells and to increase the netlist Rent parameter.[14] We plot the

Table 3
Summary of Rentian testcases with different Rent parameters[a].

Rent (input / actual)	Power (mW)	Area (um^2)	Slack (ps)
0.50 / 0.63	46.4 (100%)	39552 (100%)	−72
0.55 / 0.66	46.8 (101%)	40262 (102%)	−74
0.60 / 0.69	46.7 (101%)	40404 (102%)	−68
0.65 / 0.71	47.4 (102%)	40532 (102%)	−110
0.70 / 0.74	46.9 (101%)	40607 (103%)	−73

[a] The target clock period is 1 ns. We show both input Rent parameters to the gnl software and actual Rent parameters of the generated netlists (placement-based) in the table.

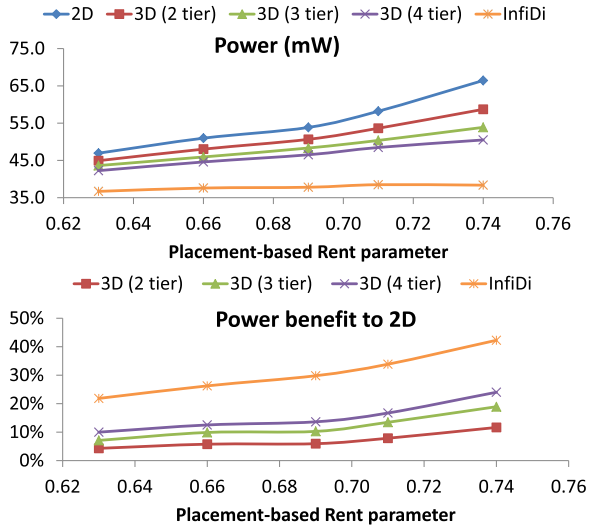

Fig. 7. Power and power benefits versus Rent parameters for the 2D and the 3D implementations with different tiers.

placement-based Rent parameters against the portion of complex cells (cells with more than three input pins) of various synthesized netlists in Fig. 9. We observe that the Rent parameters are highly correlated to the incidence (proportion) of three-input cells. This demonstrates that we can modulate Rent parameters of the synthesized netlist. However, more precise control of Rent parameters during synthesis optimization remains as a direction for future work.

In Fig. 10, we further show power (after routing) of six synthesized netlists of design JPEG which have the maximum and minimum Rent parameters (Table 4). As highlighted in blue dotted circles, we observe that although a particular netlist shows small power after synthesis (indicated by infinite dimension), due to its large Rent parameter its power can be larger with a 2D implementation. However, power penalty with a 3D implementation is smaller. This suggests that netlist synthesis should be aware of implementation dimension. For instance, a netlist with small Rent parameter is desirable for a 2D implementation; there are fewer constraints on (or, sensitivities to) Rent parameters for a 3D implementation.

4.6. SoC-level 3D benefits

The above discussion as well as many previous works on 3D wirelength benefits all focus on blocks with only standard cells (i.e., pure logic-logic integration). For example, the work of [10] applies Rent's rule-based estimation to derive wirelength distribution in 3DICs. However, our P & R results indicate that their estimated benefits (e.g., 3.9×increase in frequency) might be optimistic. Mak

[12] We constrain gnl to instantiate equal numbers of DFFX8, INVX8, BUFX8, AND2X8, NAND2X7, OR2X8, NOR2X7, NAND3X12, NOR3X13 and XOR2X8 cells in the generated netlists.

[13] The gnl software constrains the maximum delays of the generated netlists by limiting the depths of the logic cones (i.e., inserting flip-flops at the boundary of the logic cones).

[14] We synthesize JPEG with area scaled by {1x, 2x} for 2-input cells, and by {1x, 0.5x} for 3-input, 4-input and >4-input cells. An alternative way to modulate cell usage and Rent parameters during synthesis is to set a dont_use attribute for certain Liberty cells. However, the dont_use attribute cannot be assigned to NAND2, NOR2 cells in our EDA

(*footnote continued*)
tooling. In addition, setting the dont_use attribute for a group of cells might degrade synthesis solution quality due to limited available cell types.

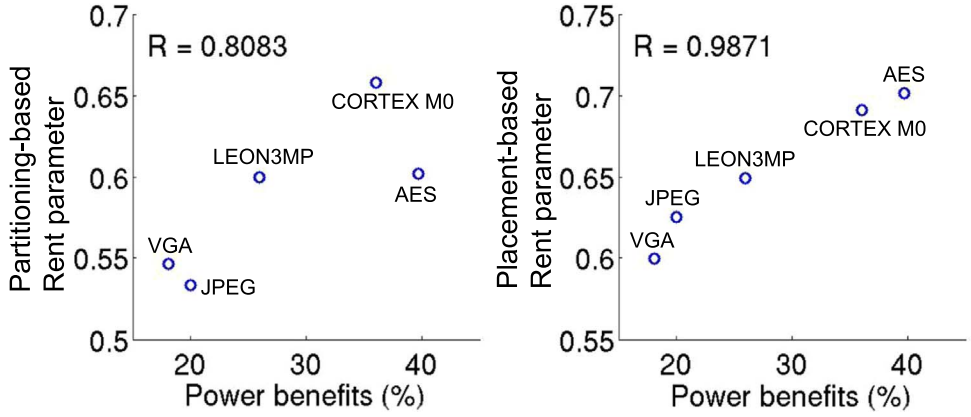

Fig. 8. Power benefits correlate with Rent parameters.

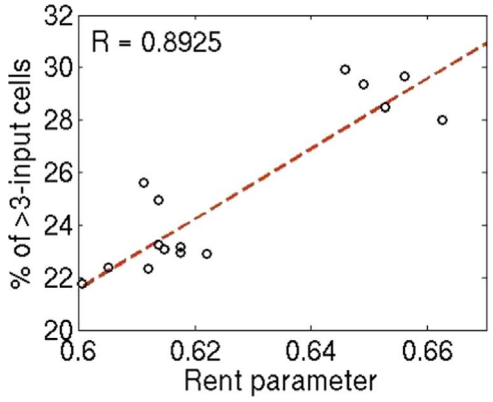

Fig. 9. Correlation between incidence of cells with >3 inputs vs. Rent parameter.

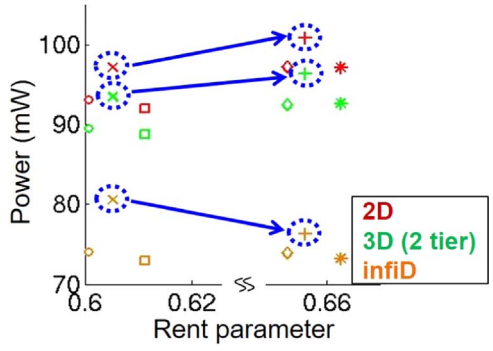

Fig. 10. Power vs. Rent parameter with Rent Modulation.

Table 4
Area scaling ratios for implementations in Fig. 10.

Implementation	Rent	2-input	3-input	4-input	>4-input
O	0.600	1	0.5	1	1
X	0.605	2	0.5	1	1
□	0.611	1	1	1	1
◇	0.653	2	1	0.5	0.5
+	0.656	2	0.5	0.5	0.5
*	0.663	1	0.5	1	0.5

et al. [26] compare 3D wirelength and 2D wirelength (where the 2D placement is simply assumed as side-by-side placement of tiers from the 3DIC) to estimate an upper bound on 3D wirelength benefits. Their results show an average of 18% wirelength reduction from 3D integration. Further, [4] uses an example to show that the maximum wirelength reduction from 3D integration is 66.7%. We show in

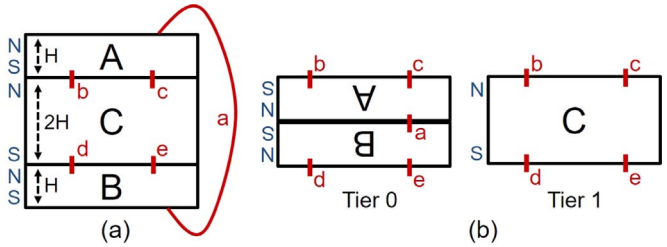

Fig. 11. (a) 2D implementation with connections (shown in red) between the north edge of Block A and the south edge of Block B, the north edge of Block C and the south edge of Block A, the south edge of Block C and the north edge of Block B, where the wirelength is at least $4{\cdot}H$. (b) 3D implementation with zero wirelength. a-e are five 2-pin nets.

Fig. 11 that for a testcase with three blocks, the wirelength reduction can be close to 100%. In the example, there are connections between the north edge of Block A and the south edge of Block B (e.g., net a), the north edge of Block C and the south edge of Block A (e.g., nets b and c), as well as the south edge of Block C and the north edge of Block B (e.g., nets d and e). As shown in Fig. 11(a), the 2D wirelength is at least $4{\cdot}H$. Fig. 11(b) shows that the 3D wirelength can be zero if the vertical wirelength is ignored. Therefore, the example with ~100% wirelength reduction indicates that there remain potential large 3DIC benefits versus 2DIC for (block-based) SoC designs. From the example in Fig. 11, we see that there are possible gains from 3D integration for designs with memory cells. Since data access latency of a large memory system is typically limited by large interconnect delays, usage of die-to-die vertical interconnects and/or side-by-side integration of a 3D stacked memory and a processor on a silicon interposer to reduce memory access latency can improve the system performance [25,35,42,44]. For example, the study of [35] shows a 47% latency reduction for a 4 MB 4-die stacked 3D SRAM array.

5. Conclusions

In this paper, we revisit previous assessments of the benefits of 3DIC implementation with respect to area, power and wirelength. Ours is the first work to estimate upper bounds on 3D power and area benefits based on the concept of *implementation with infinite dimension*. We examine several designs with our "infinite dimension" bounding methodology, and use the available area and power gaps between "best possible" 2D implementation to estimate upper bounds on 3D benefits. We further perform such 3D benefit estimation across various technologies. From our results, we observe that the 3D power and area benefits estimated in previous works (shown in Table 1) basically align within our proposed upper bounds. However, due to the design dependency of 3D benefits, benefits estimated in infinite-dimensional implementation can be small (e.g., power benefits as low as 18%) for some designs. Our study also indicates that although 3DIC

might provide relatively large benefits in power or performance, it is typically difficult for pure logic-logic 3D integration to achieve a simultaneous (10%, 10%, 10%) improvement in (performance, power, area/cost) compared to the conventional 2D implementation. Such results indicate that 3D benefits are more likely to be achieved from SoC-level and architectural-level optimizations instead of traditional P & R physical implementation optimizations. We use a simple example to show that there remain potential large 3DIC benefits versus 2DIC for (block-based) SoC designs. We also observe that inter-tier variation causes further significant reduction of available 3D power benefits.

In addition, we study design power across various dimensions and observe a correlation between design power and netlist Rent parameter. Modulation of the netlist Rent parameter during synthesis (that is, by changing the usage and distribution of fanins) suggests that a synthesis optimization that is aware of implementation dimension may be helpful for reduced power in the final physical implementation. We also note that architecture-level improvements enabled by 3D integration (e.g., larger memory bandwidth) are of course still very promising, and these are not addressed in our work.

Open directions for future research include (i) dimension-aware synthesis (i.e., synthesis for multi-tier 3D), (ii) quantitative analysis of the relationship between power benefits and Rent parameters, and (iii) architectural-level benefit exploration.

References

[1] K. Athikulwongse, D.H. Kim, M. Jung, S.K. Lim, Block-level designs of die-to-wafer bonded 3D ICs and their design quality tradeoffs, Proceedings ASP-DAC (2013) 687–692.

[2] K. Bernstein, P. Andry, J. Cann, P. Emma, Interconnects in the third dimension: design challenges for 3D ICs, Proceedings DAC. (2007) 562–567.

[3] S. Bobba, A. Chakraborty, O. Thomas, P. Batude, V.F. Pavlidis, G. De Micheli, CELONCEL: effective design technique for 3-D monolithic integration targeting high performance integrated circuits, Proceedings ASP-DAC (2011) 336–343.

[4] W.-T. J. Chan, Y. Du, A.B. Kahng, S. Nath, K. Samadi, 3DIC Benefit Estimation and Implementation Guidance from 2DIC Implementation, Proceedings DAC, pp. 1–6.

[5] K. Chang, K. Acharya, S. Sinha, B. Cline, G. Yeric, S.K. Lim, Power benefit study of monolithic 3D IC at the 7 nm technology node, Proceedings ISLPED (2015) 201–206.

[6] J. Cong, C. Liu, G. Luo, Quantitative Studies of Impact of 3D IC Design on Repeater Usage, Proceedings International VLSI/ULSI Multilevel Interconnection Conference, 2008, pp. 344–348.

[7] J. Cong, G. Luo, J. Wei, Y. Zhang, Thermal-aware 3D IC placement via transformation, Proceedings ASP-DAC (2007) 780–785.

[8] S. Das, A. Chandrakasan, R. Reif, Three-Dimensional Integrated Circuits: Performance, Design Methodology, and CAD Tools, Proceedings IEEE Computer Society Annual Symposium on VLSI, 2003, pp. 13–18.

[9] J. Deguchi, T. Sugimura, Y. Nakatani, T. Fukushima, M. Koyanagi, Quantitative derivation and evaluation of wire length distribution in three-dimensional integrated circuits using simulated quenching, Jpn. J. Appl. Phys. 45 (4B) (2006) 3260–3265.

[10] J.W. Joyner, R. Venkatesan, P. Zarkesh-Ha, J.A. Davis, J.D. Meindl, Impact of three-dimensional architectures on interconnects in gigascale, IEEE TVLSI Integr. 9 (6) (2001) 922–928.

[11] M. Jung, T. Song, Y. Wan, Y.-J. Lee, D. Mohapatra, H. Wang, G. Taylor, D. Jariwala, V. Pitchumani, P. Morrow, C. Webb, P. Fischer, S.K. Lim, How to reduce power in 3D IC designs: a case study with OpenSPARC T2 core, Proceedings CICC (2013) 1–4.

[12] M. Jung, T. Song, Y. Wan, Y. Peng, S.K. Lim, On enhancing power benefits in 3D ICs: block folding and bonding styles perspective, Proceedings DAC (2014) 1–6.

[13] D.H. Kim, K. Athikulwongse, S.K. Lim, A study of through-silicon-via impact on the 3D stacked IC layout, Proceedings ICCAD (2009) 674–680.

[14] T.-Y. Kim, T. Kim, Clock tree embedding for 3D ICs, Proceedings ASP-DAC (2010) 486–491.

[15] D.H. Kim, S. Mukhopadhyay, S.K. Lim, Through-silicon-via aware interconnect prediction and optimization for 3D stacked ICs, Proceedings SLIP (2009) 85–92.

[16] D.H. Kim, S.K. Lim, Through-silicon-via-aware delay and power prediction model for buffered interconnects in 3D ICs, Proceedings SLIP (2010) 25–31.

[17] D.H. Kim, S. Mukhopadhyay, S.K. Lim, TSV-aware interconnect distribution models for prediction of delay and power consumption of 3-D stacked ICs, IEEE TCAD 33 (9) (2014) 1384–1395.

[18] D.H. Kim, R.O. Topaloglu, S.K. Lim, Block-level 3D IC design with through-silicon-via planning, Proceedings ASP-DAC (2012) 335–340.

[19] Y.J. Lee, S.K. Lim, On GPU bus power reduction with 3D IC technologies, Proceedings DATE (2014) 1–6.

[20] S.K. Lim, Design for High Performance, Low Power, and Reliable 3D Integrated Circuits, Springer, New York, 2012.

[21] C. Liu, S.K. Lim, A design tradeoff study with monolithic 3D integration, Proceedings ISQED (2012) 529–536.

[22] Q. Liu, M. Marek-Sadowska, Wire length prediction-based technology mapping and fanout optimization, Proceedings ISPD (2005) 145–151.

[23] Q. Liu, M. Marek-Sadowska, Semi-individual wire-length prediction with application to logic synthesis, IEEE TCAD 25 (4) (2006) 611–624.

[24] Y.J. Lee, D. Limbrick, S.K. Lim, Power benefit study for ultra-high density transistor-level monolithic 3D ICs, Proceedings DAC (2013) 1–10.

[25] G.H. Loh, Y. Xie, B. Black, Processor design in 3D die-stacking technologies, IEEE Micro 27 (3) (2007) 31–48.

[26] W.K. Mak, C. Chu, Rethinking the wirelength benefit of 3-D integration, IEEE TVLSI 20 (12) (2012) 2346–2351.

[27] D.K. Nayak, S. Banna, S.K. Samal, S.K. Lim, Power, Performance, and Cost Comparisons of Monolithic 3D ICs and TSV-Based 3D ICs, Proceedings SOI-3D-Subthreshold Microelectronics Technology Unified Conference (S3S), 2015, pp. 1–2.

[28] S.H. Ok, K.R. Bae, S.K. Lim, B. Moon, Design and analysis of 3D IC-based low power stereo matching processors, Proceedings ISLPED (2013) 15–20.

[29] M.M. Ozdal, C. Amin, A. Ayupov, S.M. Burns, G.R. Wilke, C. Zhuo, ISPD-2012 discrete cell sizing contest and benchmark suite, Proceedings ISPD (2012) 161–164.

[30] S. Panth, K. Samadi, Y. Du, S.K. Lim, High-density integration of functional modules using monolithic 3D-IC technology, Proceedings ASP-DAC (2013) 681–686.

[31] S. Panth, K. Samadi, Y. Du, S.K. Lim, Design and CAD methodologies for low power gate-level monolithic 3D ICs, Proceedings ISLPED (2014) 171–176.

[32] S. Panth, K. Samadi, Y. Du, S.K. Lim, Placement-driven partitioning for congestion mitigation in monolithic 3D IC designs, IEEE TCAD 34 (4) (2014) 540–553.

[33] M. Pathak, Y.J. Lee, T. Moon, S.K. Lim, Through-silicon-via management during 3D physical design: when to add and how many?, Proceedings ICCAD (2010) 387–394.

[34] S. Priyadarshi, W.R. Davis, P.D. Franzon, Pathfinder3D: a framework for exploring early thermal tradeoffs in 3DIC, Proceedings IC Des. Technol. (ICICDT) (2014) 1–6.

[35] K. Puttaswamy, G.H. Loh, 3D-integrated SRAM components for high-performance microprocessors, IEEE Trans. Comput. 58 (10) (2009) 1369–1381.

[36] M. Rahman, R. Afonso, H. Tennakoon, C. Sechen, Power reduction via separate synthesis and physical libraries, Proceedings DAC (2011) 627–632.

[37] J.-S. Seo, I.L. Markov, D. Sylvester, D. Blaauw, On the decreasing significance of large standard cells in technology mapping, Proceedings ICCAD (2008) 116–121.

[38] T. Song, M. Jung, Y. Wan, Y. Peng, S.K. Lim, 3D IC Power Benefit Study Under Practical Design Considerations, Proceedings International Interconnect Technology Conference and Materials for Advanced Metallization Conference (IITC/MAM), 2015, pp. 335–338.

[39] T. Song, S. Panth, Y.J. Chae, S.K. Lim, Three-tier 3D ICs for more power reduction: strategies in CAD, design, and bonding selection, Proceedings ICCAD (2015) 752–757.

[40] D. Stroobandt, P. Verplaetse, J. van Campenhout, Generating synthetic benchmark circuits for evaluating CAD tools, IEEE TCAD 19 (9) (2000) 1011–1022.

[41] T. Thorolfsson, G. Luo, J. Cong, P.D. Franzon, Logic-on-Logic 3D Integration and Placement, Proceedings International 3D Systems Integration Conference (3DIC), 2010, pp. 1–4.

[42] Y. Xie, G.H. Loh, B. Black, K. Bernstein, Design space exploration for 3D architectures, ACM JETC 2 (2) (2006) 65–103.

[43] H. Xu, V.F. Pavlidis, G. De Micheli, Effect of process variations in 3D global clock distribution networks, ACM JETC 8 (3) (2012) 20:1–20:25.

[44] J. Zhao, G. Sun, G.H. Loh, Y. Xie, Optimizing GPU energy efficiency with 3D die-stacking graphics memory and reconfigurable memory interface, ACM Trans. Archit. Code Optim. 10 (4) (2013) 24:1–24:25.

[45] Cadence Innovus User Guide.

[46] MLPart. ⟨http://vlsicad.ucsd.edu/GSRC/bookshelf/Slots/Partitioning/MLPart⟩.

[47] OpenCores. ⟨http://opencores.org⟩.

[48] Synopsys Design Compiler User's Manual.

[49] Cadence Tempus User Guide.