Parametric Yield Management for 3D ICs: Models and Strategies for Improvement

CESARE FERRI, SHERIEF REDA, and R. IRIS BAHAR
Brown University

Three-Dimensional (3D) Integrated Circuits (ICs) that integrate die with Through-Silicon Vias (TSVs) promise to continue system and functionality scaling beyond the traditional geometric 2D device scaling. 3D integration also improves the performance of ICs by reducing the communication time between different chip components through the use of short TSV-based vertical wires. This reduction is particularly attractive in processors where it is desirable to reduce the access time between the main logic die and the L2 cache or the main memory die. Process variations in 2D ICs lead to a drop in parametric yield (as measured by speed, leakage and sales profits), which forces manufacturers to speed bin their chips and to sell slow chips at reduced prices. In this paper we develop a model to quantify the impact of process variations on the parametric yield of 3D ICs, and then we propose a number of integration strategies that use a graph-theoretic framework to maximize the performance, parametric yield and profits of 3D ICs. Comparing our proposed strategies to current yield-oblivious methods, it is demonstrated that it is possible to increase the number of 3D ICs in the fastest speed bins by almost 2×, while simultaneously reducing the number of slow ICs by 29.4%. This leads to an improvement in performance by up to 6.45% and an increase of about 12.48% in total sales revenue using up-to-date market price models.

Categories and Subject Descriptors: B.7.1 [**Integrated Circuits**]: Types and Design Styles—*Advanced technologies*; B.8.2 [**Performance and Reliability**]: Performance Analysis and Design Aids

General Terms: Design, Performance

Additional Key Words and Phrases: 3D integration, process variations, performance, leakage, yield management

ACM Reference Format:

Ferri, C., Reda, S., and Bahar, R. I. 2008. Parametric yield management for 3D ICs: Models and strategies for improvement. ACM J. Emerg. Technol. Comput. Syst. 4, 4, Article 19 (October 2008), 22 pages. DOI = 10.1145/1412587.1412592 http://doi.acm.org/10.1145/1412587.1412592

An earlier version of this article appeared in *the International Conference on Computer-Aided Design* 2007. This journal version includes expanded, rewritten coverage of the technical material; a new, generic formulation to improve the parametric yield for 3D ICs with arbitrary number of die; and a new section (and experimental results) on leakage modeling and leakage-constrained parametric yield improvement.
This work was partially supported by a gift from Qualcomm Corporation.

Authors' addresses: C. Ferri, S. Reda, R. I. Bahar, Division of Engineering, Brown University, Providence, RI 02912; email: {cesare_ferri, sherief_reda, iris_bahar}@brown.edu.

1. INTRODUCTION AND MOTIVATION

Three-dimensional (3D) integrated circuits (ICs) with through-silicon vias comprise an exciting new technology that will increase the functionality, scale of integration, and performance of ICs [Banerjee et al. 2001; Topol et al. 2006]. Increasing the scale of integration is particularly attractive considering that optical lithography is approaching its natural limits as predicted by the International Technology Roadmap for Semiconductors (ITRS) [ITRS 2008]. In 3D ICs, multiple die or device layers are integrated and interconnected with *through-silicon vias* (TSVs) (also known as *vertical interconnects*), as shown in Figure 1. The theoretical possibility of integrating tens of die in a 3D IC can usher in a new era of computational platforms with capabilities that are far beyond what is currently possible.

In order to reap the full benefits of 3D integration, yield loss is one of the greatest challenges that has to be met [Banerjee et al. 2001; Topol et al. 2006; Patti 2006]. The fabrication yield of integrated circuits is divided into two categories: *functional yield* and *parametric yield*. Functional yield is the number of fabricated functionally good die with no detected manufacturing defects. Parametric yield is the number of functional die meeting the required speed and power specifications [Rao et al. 2005]. Process variations are the main contributors to the loss of parametric yield [Bowman et al. 2002; Orshansky et al. 2002; Borkar et al. 2003; Raj et al. 2004; Datta et al. 2006; Marculescu and Talpes 2005; Bhardwaj et al. 2006]. The theoretical possibility of integrating tens of interconnected die is practically limited by the yield of the 3D fabrication process. Yield loss, whether due to functional or parametric mechanisms, can occur either during the fabrication of the individual planar die or during the process of integrating and interconnecting the different die together in the 3D IC stack [Banerjee et al. 2001; Reif et al. 2002; Davis et al. 2005; Topol et al. 2006; Patti 2006].

The objective of this article is to model the parametric yield of 3D ICs as well as to provide integration strategies that maximize the parametric yield. More specifically, the contributions of this article are as follows.

(1) This work is the first to examine the impact of process variation on the performance and parametric yield of 3D ICs. We formulate the general problem of optimizing the parametric yield in 3D integration under the presence of general process variations.
(2) Using a processor as a 3D IC example, we model the impact of process variations on both the CPU and L2 cache die, and then model the outcome of 3D integration on the overall performance of the processors.
(3) This work is first to propose 3D integration strategies that maximize the parametric yield of 3D ICs using a number of criteria including performance, leakage, and realistic price models.
(4) Using extensive simulations and realistic binning strategies, we show that the proposed strategies increase the number of 3D processors in the fastest bins by almost 2×, while simultaneously reducing the number of slow processors by 29.4% in comparison to current integration techniques. Our

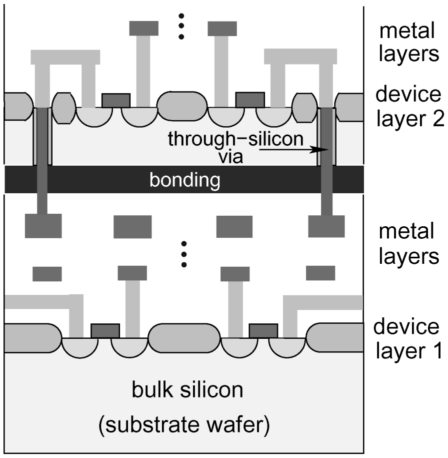


Fig. 1. A simple illustration of a 3D IC with TSVs (via first, front to back integration). More die layers are assumed stacked but not drawn.

strategy also leads to an improvement in 3D processor performance (as measured by MIPS) by up to 6.45% and an increase of about 12.48% in total sales revenue using up-to-date market-price models.

The organization of this article is as follows. Section 2 gives an overview of the necessary background for this work. In Section 3, we show how to model the performance of 3D ICs, using a 3D processor as an example, under the presence of process variations. In Section 4, we propose a number of integration strategies to maximize the parametric yield of 3D ICs. Section 5 gives an extensive set of experimental results supporting our methodology, and finally Section 6 summarizes the main results of this article.

2. BACKGROUND AND MOTIVATION

In this section we give the technical background that provides the motivation and context for this work. In particular, we provide an overview of the main 3D fabrication methods and discuss the main benefits of 3D integration. Then we discuss the impact of the fabrication method on the final yield. We also provide a brief overview of the impact of process variations in 2D ICs which ultimately leads to the variations in 3D ICs.

Benefits of 3D Integration. 3D ICs allow the creation of new systems that are currently not feasible by planar fabrication technology. Using a 3D approach allows the integration of dissimilar technologies to create highly-interconnected hybrid chips that include memory, logic, optical, RF, and analog components. Besides improved functionality and system scaling capabilities, 3D integration also promises to replace long 2D interconnects by short TSV-based vertical interconnects [Banerjee et al. 2001; Davis et al. 2005; Topol et al. 2006]. Long (or global) 2D interconnects have large delays [Davis et al. 2001] and require an increasing number of repeaters to appropriately buffer them [Saxena et al. 2004]. By transforming long 2D interconnects into short TSVs with less capacitive and resistive loading, the system delay is improved [Topol et al. 2006]. Reducing long interconnect delay is especially important for processors, as they continuously access memory subsystems. With 3D integration, processors can cut down the memory-access time, which improves the overall system performance. The quantification of this improvement has been the subject of a number of recent works [Zeng et al. 2005; Liu et al. 2005; Jacob et al. 2005; Tsai et al. 2005; Li et al. 2006; Xie et al. 2006].

3D IC fabrication Techniques. There are four main manufacturing steps during 3D IC fabrication: thinning, alignment, bonding, and through-silicon via fabrication [Burns et al. 2006; Reif et al. 2002; Benkart et al. 2005; Beyne 2004; Davis et al. 2005; Scheiring 2004; Topol et al. 2006; Patti 2006]. *Thinning* involves removing the bulk silicon of a wafer, bringing it to only a few tens microns of thickness. *Alignment* places the different wafers or die of a 3D IC on top of each other, with their faces (or backs) aligned within some allowed tolerance. This tolerance imposes a limit on the size and pitch of the through-silicon vias. *Bonding* fuses the different wafers and/or die together. *Through-silicon via fabrication* creates the vertical interconnects that are required for signal communication between various parts of the design in the 3D ICs. The via creation step can also be carried out before or after the bonding step [Baliga 2004].

While failure in any of these four steps impacts the yield of 3D ICs, the method of bonding is typically the most critical step [Scheiring 2004; Topol et al. 2006; Fukushima et al. 3035]. There are currently three different bonding technologies that offer different trade-offs in production yield, flexibility in die size, and production throughput [Burns et al. 2006; Reif et al. 2002; Benkart et al. 2005; Davis et al. 2005; Topol et al. 2006; Patti 2006].

(1) *Wafer-to-Wafer Bonding*. This method results in the lowest yield of all bonding methods, since it offers no way to filter out the bad die before integration. It also offers no flexibility in choosing how to “pick and place” the die to optimize both the parametric and functional yield. This method also requires all die to be of the same size. Its main advantage is in high production throughput and high TSV density.
(2) *Die-to-Wafer Bonding*. This method uses a substrate wafer to integrate diced die on top of it. It has a high yield, as it is possible to identify and only use the good die during integration. The method is flexible with different die sizes and has a good production throughput.
(3) *Die-to-Die Bonding*. This method offers similar high yield and flexibility as in die-to-wafer bonding, but suffers from low production throughput.

Comparing the three bonding methods, it is typically concluded that die-to-wafer bonding is the most promising for future 3D integration [Scheiring 2004; Topol et al. 2006; Fukushima et al. 3035].

Impact of Process Variations in 2D ICs. Since the main components of a 3D IC are 2D die, it is natural that the same physical phenomena, manufacturing defects, and process variations that reduce the yield of 2D ICs will also impact 3D ICs. Thus, it is important to understand these phenomena in 2D ICs before generalizing them to 3D ICs. Process variations change the electrical parameters of ICs (e.g., speed and leakage) from the original estimates of the designers. Process variations can heavily impact the frequency at which a processor can be clocked, the total power dissipation due to leakage current [Bowman et al. 2002; Borkar et al. 2003; Marculescu and Talpes 2005; Humenay et al. 2006; Kim et al. 2006], and the relative access time of the memory subsystem [Grossar et al. 2006; Meng and Joseph 2006]. Semiconductor foundries typically categorize chips according to their performance by *speed-binning* them and assigning them to appropriate price points [Cory et al. 2003; Datta et al. 2006]. Improving the parametric yield is concerned with optimizing the values of the electrical parameters of chips in order to achieve overall good performance and profits [Rao et al. 2005; Datta et al. 2006]. Also, process variations can increase leakage current by up to 20× [Borkar et al. 2003; Kim et al. 2006]; it is crucial to minimize such leakage in chips that are embedded in low-power devices. In this case, the binning strategy could be driven entirely by leakage constraints, where high-leakage chips are essentially discarded.

Yield loss, whether functional or parametric, is considered one of the bottlenecks that need to be overcome to bring 3D technology “from the lab to the fab” and the marketplace [Baliga 2004]. Despite its importance, the problem of yield improvement of 3D ICs has been the least investigated in the literature. A number of recent efforts [Banerjee et al. 2001; Topol et al. 2006; Patti 2006] point to the importance of functional yield management of 3D ICs. This work is the first to investigate the problem of process variation modeling and parametric yield improvement in 3D ICs.

3. PARAMETRIC YIELD MODELING

The objective of this section is to model or quantify the impact of process variations on the parametric yield of 3D ICs. Such modeling is more complex than in 2D ICs, as different die that belong to the same 3D IC are fabricated on separate wafers and then integrated and interconnected with TSVs. Thus, to model the impact of process variations on 3D ICs, it is necessary to first model the impact of process variation on the individual die and then model the interplay of process variations on the different die composing a 3D IC.

In 2D ICs, process variations can be categorized as *intradie variations*, which affect subparts of a single chip, and as *interdie variations*, which affect the performance and power parameters of different chips [Bowman et al. 2002]. The overall impact of intra- and interdie variations is that they lead to considerable discrepancies in the performance of fabricated chips. The distribution of chips as a function of performance typically exhibits a Gaussian-like form [Orshansky et al. 2002; Bowman et al. 2002], where the mean and standard

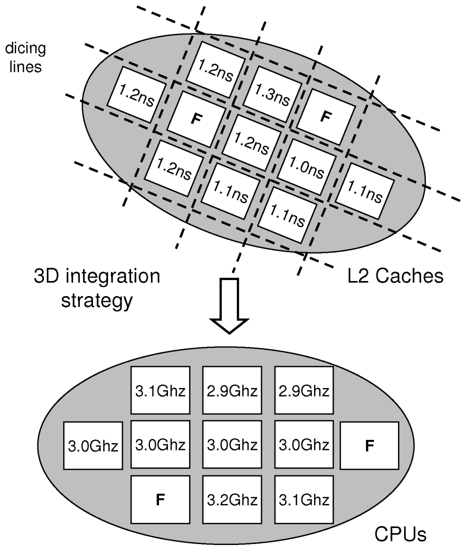


Fig. 2. Modeling the impact of process variations on 3D processors. **F** indicates a faulty die. The number inside each die represents its speed as measured by testing before 3D integration.

deviation of the distribution are functions of the intradie and interdie variations, respectively.

The final result of the interaction of process variations on different die depends on the functionality and the interface of different die in the 3D stack. Consider, for example, the 3D processor given in Figure 2. The upper wafer holds a set of die, say L2 cache or memory die. The die have been diced and tested, and the faulty ones have been identified (labeled with **F**) and the good ones have been labeled with their speed and leakage (only speed is shown in Figure 2).[1] The same testing and labeling procedure has been carried out for the substrate wafer containing the central processing unit (CPU) die. The question we now seek to answer is: What is the impact of the process variations in the individual 2D die on the overall 3D IC performance and leakage?

To quantify the impact of process variations on the overall performance of the 3D processor, we choose the popular *MIPS (millions of instructions per second)* as the performance index. For a given pair (i, j) of CPU i and L2 cache j, we compute the MIPS in the following way. We first calculate the L2 latency, $L_{i,j}$, in terms of CPU cycles, as

$$L_{i,j} = \left\lceil \frac{\text{L2 } j \text{ accesstime}}{\text{CPU } i \text{ cycle period}} \right\rceil . \tag{1}$$

[1] 3D technology offers unique ways to address the reliability issues of 3D memories. The basic assumption is that the higher bandwidth guaranteed by a 3D structure (because of superior interconnect density) simplifies enormously the operations of remapping faulty memory cells. In particular, Patti [2006] (also commercialized by Tezzaron) proposes continuously monitoring in the background the actual state of each memory cell, and repairing on-the-fly the errors by replacing the cells with nonfaulty ones.

While the access time and core frequency may display wide variations, the ceiling rounding, by the $\lceil\cdot\rceil$ operator, reduces the number of distinct latency values. Note that the L2 latency varies from a minimum of $L_{\min} = \lceil\text{min L2 access time} \times \text{min CPU frequency}\rceil$ to a maximum of $L_{\max} = \lceil\text{max L2 access time} \times \text{max CPU frequency}\rceil$.

The impact of cache latency on performance depends on the particular application executing on the processor. A memory-intensive application will be heavily impacted by large values of memory latency in comparison to a processing-intensive application. To obtain an accurate estimation of the overall speed of the 3D processor, the cache latency values need to be fed to an architectural simulator to compute the actual *cycles per instruction (CPI)* using typical benchmark applications. The fact that there are only a few possible values for the L2 latency drastically reduces the number of architectural simulations that need to be carried out. Finally, the MIPS of the 3D chip composed of CPU i and cache j is simply the clock frequency of i multiplied by the CPI of the pair.

While modeling the final performance of a 3D IC requires design knowledge (e.g., the processor and its application programs), modeling the final leakage of a 3D IC is more straight forward. The final leakage will be the sum of the leakage currents of the constituent die in the 3D IC while taking into account the spatial and temporal variations in temperature of the 3D IC [Im and Banerjee 2000; Loi et al. 2006].

We note that while we extensively use the 3D processor as a specific potential 3D IC, our general parametric yield modeling and improvement methodology is still applicable to other 3D IC designs. For example, we sketch an outline for modeling two other potential 3D ICs: 3D field programmable gate arrays (FPGAs) and 3D embedded systems-on-a-chip (SoC).

—*3D FPGAs*. Heterogeneous 3D FPGAs could be one of the great applications of 3D technology. One can envision a stack of die, which has reconfigurable logic in one set of die, hard IPs (e.g., processors or DSPs) on another, and memory on a third set. Consider a 3D FPGA, where a multicore system on one die is interfaced to another die of reconfigurable computing. The reconfigurable logic provides the necessary fabric to accelerate key software routines. The speed of the system depends on the rates by which the cores call the reconfigurable logic, which in turn depends on the workloads running on the cores. As a first-order analysis, the overall delay can be considered equal to processor cycle delay$+\frac{1}{callingrate}\times$ reconfigurable logic delay.

—*Embedded SoC 3D ICs.* Consider an embedded 3D IC that is designed for multimedia applications, where a critical component of the IC is dedicated to computing the fast Fourier transform (FFT). An FFT computational system involves a good number of pipeline stages [Baas 1999]. To reduce the communication overhead between the stages, it is advantageous to place them, especially those that use global interconnects in the FFT's butterfly structure [Das et al. 2004; Baas 1999], on separate die. In this case, the maximum operating frequency of the system is determined by that pipeline stage with the largest delay while considering the impact of process variations on the individual pipeline stages located in the different die.

4. STRATEGIES FOR IMPROVING THE PARAMETRIC YIELD OF 3D ICs

While wafer-to-wafer bonding dictates the outcome of integrating different wafers, die-to-wafer and die-to-die integration offer flexibility that we propose to exploit by devising 3D integration strategies that maximize the parametric yield. The general problem of optimizing the parametric yield of 3D ICs under process variations can be stated as follows.

The 3D Parametric Yield Maximization Problem for 3D ICs. Given K different wafers (or wafer lots), each with identically N die, yet the die are parametrically different due to process variations, find an integration assignment strategy that maximizes the total parametric yield of the N produced 3D ICs, where each IC is composed of K stacked die.

The outline solution for this problem is as follows.

(1) Model the impact of the process variations on both the speed and leakage on each die for all K wafers.
(2) Model the performance of the 3D system (composed of K different dies) for every possible N^K 3D IC combination.
(3) From the N^K possible combinations, find the N combinations that maximize the total parametric yield (as measured by performance, leakage, or revenue) such that each die is assigned to exactly one 3D IC package.

The problem is obviously electrically and combinatorially challenging. First, the impact of process variations on the electrical properties (speed and leakage) have to be modeled for each die and for each possible 3D combination, and second, the N 3D IC combinations that maximize the total parametric yield have to be computed and selected. For the case of three or more integrated die, one can prove that maximizing the parametric yield for 3D ICs is NP-hard, by reducing the classical NP-hard 3-D matching problem [Garey and Johnson 1979] to it. A more tractable version of the problem is possible in the case of two die (i.e., $K = 2$) where, for example, the first wafer holds processor logic and the second wafer holds the processor L2 cache (or memory in general).

We propose a number of strategies that control and improve the parametric yield of 3D ICs. First, we focus on improving the parametric yield as measured by the speed or performance of the 3D package. Later, we will focus on yield as measured by sales profit or leakage.

4.1 Assignment Strategies for Maximizing Performance

The proposed strategies vary in their ability to optimize the parametric yield, and also in their computational complexity.

—*Random-Random (RR) Assignment.* In this naive strategy, the 3D integration process is oblivious to parametric yield and assigns CPUs and L2 caches randomly to form the 3D processor chips. This strategy can be used as a baseline to compare other strategies against.

—*Fast-Fast (FF) Assignment.* In this strategy, CPU die are sorted in descending order (fastest first) according to their tested speed (CPU frequency), and then L2 cache die are sorted in ascending order (fastest first) according to their tested speed (access time). Then the 3D chips are constructed by matching the CPUs and L2 caches in order. This strategy starts pairing the fastest CPUs and L2 caches together and ends pairing the slowest CPUs and caches together. This strategy attempts to obtain the fastest possible 3D processor chips (at the cost of producing the slowest possible 3D chips). This strategy is easily computed in $O(N \log N)$ runtime, and for K die stacks, it is computable in $O(KN \log N)$ runtime.

—*Fast-Slow (FS) Assignment.* In this strategy, CPU die are sorted in descending order (fastest first) according to their tested speed (CPU frequency), and then L2 caches are sorted in ascending order (slowest first) according to their tested speed (access time). Then the 3D chips are constructed by matching the CPUs and L2 caches in order. This strategy starts pairing the fastest CPUs with the slowest L2 caches together and ends pairing the slowest CPUs with the fastest caches together. This strategy attempts to increase the number of processors with medium speed. It can also be helpful if leakage is the main factor driving the integration strategy because it integrates low-leakage die with high-leakage die, decreasing the overall 3D IC leakage. This strategy is easily computed in $O(N \log N)$ runtime, and for K die stacks, it is computable in $O(KN \log N)$ runtime.

—*Optimal (OPT) Assignment.* To find the optimal integration strategy, we propose an integer linear program (ILP) that maximizes the parametric yield for any number of die in the 3D stack. Let $x_{i_1,i_2,\ldots,i_K}$ denote a binary variable that is true when die $i_1 \in \{1,\ldots,N\}$ from wafer 1, die $i_2 \in \{1,\ldots,N\}$ from wafer 2, …, and die $i_K \in \{1,\ldots,N\}$ from wafer K are integrated into a 3D IC. Let $Y_{i_1,i_2,\ldots,i_K}$ be a constant that gives the parametric yield of the 3D IC formed from $i_1, i_2, \ldots,$ and i_K as defined by the speed, leakage, direct revenue, or a combination of these. Given K wafers with N die, the parametric yield maximization problem can be formulated into the ILP as

$$\max \sum_{i_1=1}^{N} \cdots \sum_{i_K=1}^{N} Y_{i_1,i_2,\ldots,i_K} \times x_{i_1,\ldots,i_K}, \tag{2}$$

such that there are exactly N produced 3D ICs

$$\sum_{i_1=1}^{N} \cdots \sum_{i_K=1}^{N} x_{i_1,\ldots,i_K} = N, \tag{3}$$

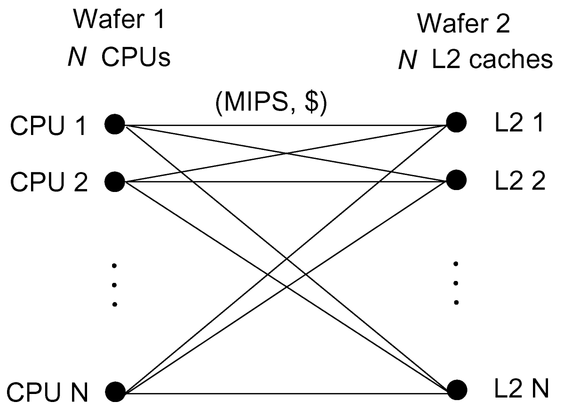


Fig. 3. Optimal assignment of CPUs to L2 caches to generate 3D processor chips that maximize the total parametric yield as measured by the total system MIPs.

and each die on any wafer participates in exactly one 3D IC

$$\begin{aligned}
\forall i_1 \in \{1,\ldots,N\}: &\quad \sum_{i_2=1}^{N}\cdots\sum_{i_K=1}^{N} x_{i_1,\ldots,i_K} = 1\\
&\ldots\\
\forall i_j \in \{1,\ldots,N\}: &\quad \sum_{i_1=1}^{N}\cdots\sum_{i_q=1,q\neq j}^{N}\cdots\sum_{i_K=1}^{N} x_{i_1,\ldots,i_K} = 1\\
&\ldots\\
\forall i_K \in \{1,\ldots,N\}: &\quad \sum_{i_1=1}^{N}\cdots\sum_{i_{K-1}=1}^{N} x_{i_1,\ldots,i_K} = 1
\end{aligned}$$

For the case of $K = 2$, it is possible to find the optimal solution to the ILP program in polynomial runtime using a graph-theoretic framework. In this case, *vertices* represent the die, *edges* represent the possible 3D ICs, and *edge costs* represent the yield (speed or revenue) value of the possible ICs. Thus, we construct a bipartite graph, given in Figure 3, with $2N$ vertices representing the N CPU die and the N L2 cache die, and N^2 edges, where each edge is labeled by the MIPS of the 3D processor produced from the CPU and L2 cache die that are its end-points. The optimal assignment strategy involves finding the N CPU/L2 pairs that maximize the total MIPS, and such that each CPU or L2 cache participates in only one 3D IC. The optimal assignment can be found by computing the maximum graph matching or assignment in the bipartite graph. This can be computed in polynomial $O(N^3)$ runtime using the classical Hungarian algorithm [Kuhn 1955; Munkres 1957].

The performance of a 3D system determines its speed bin and consequently its price. This is described in the next subsection.

4.2 Strategies to Maximize Sales Profits

Chip manufacturers are ultimately interested in maximizing sales profits. Processors with higher performance (measured by MIPS) are naturally sold at

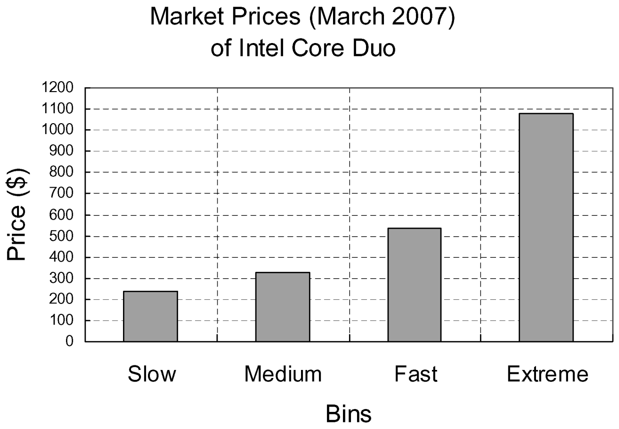


Fig. 4. Market prices (according to pricegrabber.com) of Intel Core Duo as of March 2007.

higher prices. The difference in price is correlated with the number of available supply chips. Since process variations produce chips with Gaussian-like distributions, it is expected that there are very few chips with extremely high or extremely low performance and the majority of chips have a performance around some average value. This leads to a nonlinear relationship between the performance and price of the chip. For example, the market values of Intel Core Duo processors (according to pricegrabber.com) for its different four speed bins are given in Figure 4. The plot shows an exponential trend for the price. The price of extreme processors are almost double the price of fast processors, which are in turn double the price of the slow ones. We also note that binning can be partially driven by the market dynamics of supply and demand. Fast chips can be either sold as slow or fast, depending on the market demand and supply forces. However, slow chips can only be binned as slow. This asymmetric situation means that fast chips offer greater flexibility in meeting market requirements.

Our proposed fast-fast and optimal-assignment strategies are designed to increase the number of fastest 3D chips (as we will confirm in Section 5). Thus, this is likely to lead to a significant increase in total sales profits and flexibility according to the market price model. It is also possible to directly derive our optimal assignment strategy, described in Section 4, using the dollar values of the 3D system rather than using the MIPS value. In this case, we can substitute the MIPS label of each edge in Figure 3 by the corresponding dollar value and find the optimal assignment strategy as described earlier.

4.3 Leakage-Aware Assignment Strategies

As described in Section 2, chips with the highest performance (or smallest delay) are likely to produce chips with the highest leakage current. Thus, optimal, or fast-fast, assignment strategies can produce 3D chips with excessive leakage power, since they produce systems with the highest performance. This excessive leakage power can be problematic for chips that are targeted for low-power mobile devices. Thus, we seek to modify our 3D integration strategies to take into account leakage power to improve the parametric yield. There could be two possible modifications, depending on the importance of leakage current.

(A) *Leakage-Constrained Integration.* In this approach, we still use performance-driven integration but under the constraint that a 3D IC should never exceed a certain leakage threshold. This modification can be accommodated in the proposed graph-theoretic approach as follows. After we generate the bipartite graph shown in Figure 3, we delete any edges that correspond to CPU/L2 cache pairs that generate leakage current above the allowed leakage threshold. The new graph is then used to calculate the optimal assignment. The optimal assignment algorithm automatically results in performance-optimized 3D processor chips that produce leakage current/power below the constraining threshold.

(B) *Leakage-Driven Integration.* In this approach (which is more suitable for ultralow-power mobile 3D ICs), we completely drive the 3D integration process by leakage. Given a stringent leakage threshold, we label every edge of Figure 3 with the sum of the leakage of its end-points if and only if this sum is below the given threshold; otherwise, we label the edge with a high prohibitive cost (ideally, ∞).

5. EXPERIMENTAL RESULTS

In this section we quantify the impact of our 3D integration strategies on the parametric yield and profits of 3D ICs. A key input to our models and strategies is the basic speed and leakage test results from the 2D ICs that will form the 3D ICs. Since such test results are not available, we estimate these numbers through simulations of realistic CPU and cache hardware models. We use the following tools.

—SPICE to calculate the delay of CPUs under the presence of process variations using 70nm technology [PTM 2008].

—CACTI (version 4.2) [Wilton and Jouppi 1996] and PRACTICS (version 1.0) [Zeng et al. 2005] to calculate the access time of L2 caches using vertical interconnects in 3D chips.

—SimpleScalar (version 3.0) [Burger and Austin 1997] to model the performance (as measured by cycles per instruction CPI) of 3D processors, given the underlying CPU frequency and the L2 cache-access time with vertical interconnects.

—the matching code by Cook and Rohe [1999] to implement the optimal 3D assignment strategy to maximize the parametric yield.

Our tool-chain flow, given in Figure 5, starts by modeling the impact of process variations on the speed and leakage of $N = 100$ CPUs and the L2 cache die. Then an architectural simulator, SimpleScalar, is used to calculate the performance of the potential 3D processors composed of the different CPUs and L2 die. This information, together with the leakage current, is used to construct a bipartite graph as outlined in Section 4. This graph is then fed, with

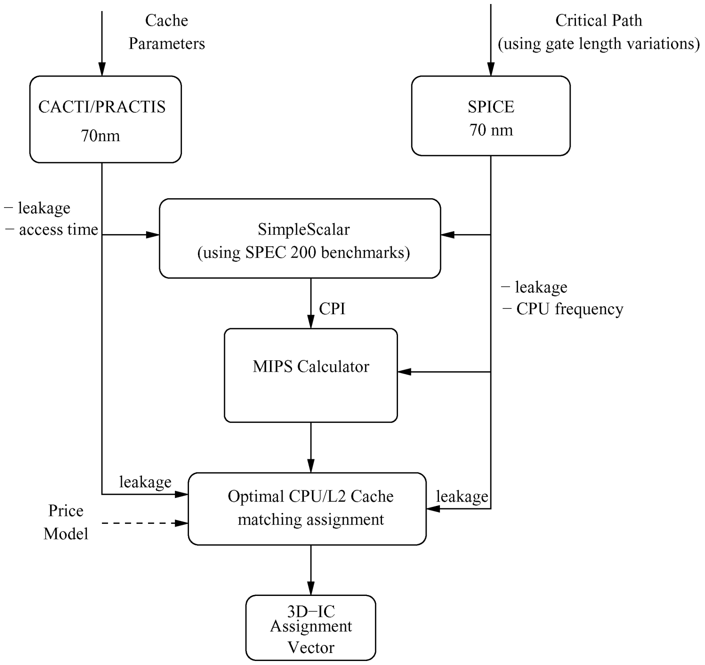


Fig. 5. The tool chain required to model and evaluate our strategy.

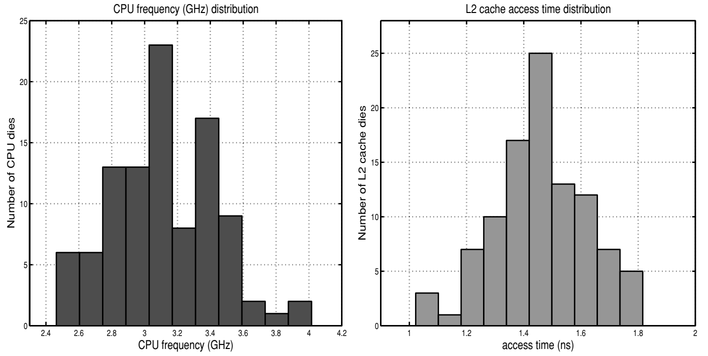
market-price models, to the optimal matching module to find the integration assignment that maximizes the parametric yield and profits. In the following subsections, we describe each tool and step in detail.

5.1 CPU Setup

We quantify the impact of process variations on the performance and power of 2D CPUs by simulating with SPICE a typical CPU critical path (i.e., a chain of 9 NAND gates representing the CPU pipeline flow [Bowman et al. 2002]). We use the 70nm Berkeley predictive technology model for all simulations [PTM 2008]. To model the impact of interdie process variations, we generate 100 critical-path SPICE netlists, where the gate length of each is drawn from a Gaussian distribution with a mean of 70nm and standard deviation of 5.07% (leading to ±7nm maximum variations). We then execute SPICE on each netlist and record the delay and leakage current consumed. The frequency of a CPU is the reciprocal of the critical-path delay. We plot the distribution of CPU frequencies (GHz) obtained from SPICE simulations in Figure 6(a). Table I gives the maximum, minimum, average, and standard deviation of the distribution of CPU frequencies and leakage. The distribution of CPUs has a standard deviation of 10.33% with a mean of 3.12 GHz.

5.2 L2 Cache Setup

Assuming a cache configuration of 2 MB at the 70nm technology node, we calculate the cache-access time using PRACTICS [Zeng et al. 2005], which is a



(a) CPUs frequency (Ghz) distribution as produced by modeling critical-path variations on the CPU delay using SPICE

(b) L2 cache-access time (ns) distribution as produced from modeling process variations using CACTI and PRACTICS

Fig. 6. Impact of process variations on CPU frequency and L2 cache access time.

tool for predicting the access time of L2 caches using vertical interconnects in 3D ICs. To obtain our cache leakage variation numbers, we assume normally distributed gate-length variations that vary ±7nm around the 70nm nominal gate length. Using PRACTICS to simulate the latency of caches with such variations resulted in cache-access time variations with a standard deviation of 11.06%. Figure 6(b) plots the resulting distribution for the L2 cache-access time (ns). The main statistics of the L2 access time and leakage distribution are reported in Table I.

5.3 3D Processor Performance Modeling

With the computed CPU frequency vector (100 values in GHz) and the L2 access time vector (100 values in ns), it is possible to calculate the L2 access time in terms of CPU cycles, namely $\lceil \frac{\text{L2 access time}}{\text{CPU cycle period}} \rceil$, for every possible pair of CPU and L2 cache. While the number of different CPU frequencies and cache-access times could be large due to process variations, the number of distinct different cache-access cycles is much fewer (e.g., they vary between 3 to 8 cycles). The newly computed values for access cycles are used as configuration parameters for the SimpleScalar simulator (which requires cache-access time expressed in CPU cycles) to simulate the performance of every possible CPU/L2 3D chip combination.[2] Next, we run a suite of six SPEC 2000 benchmarks [SPEC 2000] and compute the average cycles per instruction (CPI) over the six benchmarks: three integer benchmarks (gcc, parser, gzip) and three floating-point applications (mgris, apsi, equake). CPI results are given in Table II. We then use the CPI and clock frequency values to calculate the MIPS of every possible CPU/L2 3D processor.

[2] We use the following parameters for simulation: (1) two-way, three-cycle L1 cache of 16Kbyte; (2) eight-way 2MB L2 cache; (3) main memory latency of 50 cycles; and (4) the decode/issue/commit width is 4 issue.

Table I. Impact of Process Variations on the Speed of CPU and L2 Cache Dies

Parameter	CPU		Parameter	L2 Cache	
frequency	max	4.01GHz	access time	max	1.81ns
	average	3.12GHz		average	1.46ns
	min	2.46GHz		min	1.02ns
	std dev.	10.33%		std dev.	11.06%
leakage	max	17.8W	leakage	max	13.4W
	average	5.09W		average	8.22W
	min	1.93W		min	4.57W
	range ($=\frac{\text{max}}{\text{min}}$)	5.21×		range ($=\frac{\text{max}}{\text{min}}$)	2.93×

Table II. CPI Reported for Different L2 Cache-Access Cycles $\left(\left\lceil \frac{\text{L2 access time}}{\text{CPU cycle period}} \right\rceil\right)$

L2 Latency (cycles)	Bench	CPI	Avg CPI	L2 Latency (cycles)	Bench	CPI	Avg CPI
3	apsi	0.614	0.734	6	apsi	0.614	0.847
	equake	0.785			equake	0.905	
	gcc	1.031			gcc	1.556	
	gzip	0.577			gzip	0.594	
	mgrid	0.548			mgrid	0.547	
	parser	0.850			parser	0.863	
4	apsi	0.614	0.798	7	apsi	0.615	0.873
	equake	0.865			equake	0.927	
	gcc	1.330			gcc	1.669	
	gzip	0.585			gzip	0.599	
	mgrid	0.543			mgrid	0.559	
	parser	0.851			parser	0.868	
5	apsi	0.614	0.819	8	apsi	0.615	0.899
	equake	0.875			equake	0.955	
	gcc	1.441			gcc	1.786	
	gzip	0.585			gzip	0.604	
	mgrid	0.546			mgrid	0.561	
	parser	0.855			parser	0.876	

L2 access times and CPU clock periods are taken from the data of Figure 6.

5.4 Evaluation of 3D Integration Strategies

With the modeled CPU frequency, L2 access time, and processor MIPS it is possible to evaluate the effectiveness of our different 3D integration strategies on the parametric yield as measured by the performance of the 3D processor. Given the CPU frequency and L2 access-time distributions of Figure 6, we compute the MIPS distributions of 3D processors produced by different assignment strategies (RR, FF, FS, and OPT). We report the performance in terms of average, maximum, and minimum MIPS in Table III. The results of Table III demonstrate that the optimal-assignment strategy and the fast-fast strategy produce systems with the maximum MIPS; however, the optimal strategy has the highest average MIPS of all strategies. Compared to the performance-oblivious strategy (the random-random strategy), the optimal-assignment strategy produces

Table III. Impact of Different 3D Integration Strategies on Statistical Performance Parameters of 3D Processor Chips

Strategy	Max MIPS	Average	Min MIPS	Δ MIPS (%)
Fast-Fast	4902.62	3810.41	3006.78	63.04%
Fast-Slow	4465.68	3784.63	3221.22	38.63%
Optimal	4903.00	3855.00	3138.00	56.25%
Random-Random	4606.61	3790.17	3082.71	49.93%

We calculate Δ MIPS (%)= $\frac{\text{Max MIPS–Min MIPS}}{\text{Min MIPS}}$.

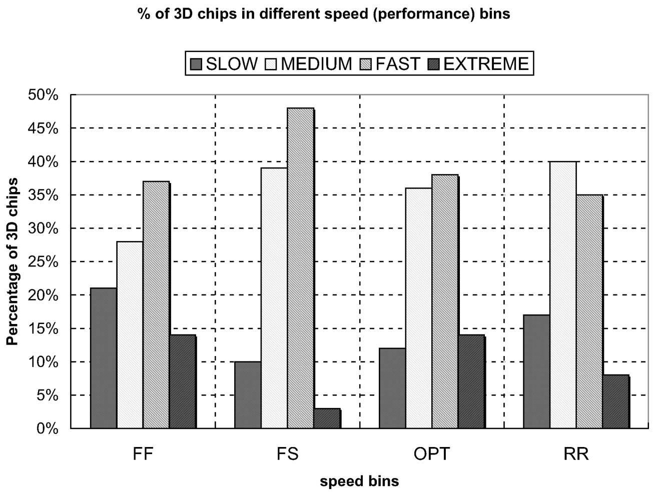


Fig. 7. Impact of process variations on 3D processor performance as measured by MIPS, using the different proposed 3D integration assignment strategies. Four performance or speed bins are used: slow, medium, fast, and extreme.

a system with better performance by up to 6.49%, with an average improvement of 1.71%.

The processor distributions produced from different integration strategies are also given in Figure 7. To create the figure, we use the RR processor distribution to designate four performance bins, extreme, fast, medium, and slow, using Matlab's histogram function. The four bins mimic those of Intel processors as we described earlier in Subsection 4.2. We use the bin boundaries of the RR strategy as the bin boundaries of other 3D integration strategies. This way we guarantee a fair comparison for the different strategies. From the data, we draw the following observations.

- The OPT and FF strategies produce almost twice the number of extreme processors compared to other strategies.
- While OPT and FF produce the same number of extreme processors, OPT reduces the number of processors in the slow bin by almost half compared to FF. Note that FF produces the highest number of CPUs in the slow bin.
- FS produces a large number of CPUs in the medium and fast bins, but produces the fewest number of CPUs in the extreme bin.

Table IV. Impact of Different 3D Assignment Strategies on Number of Processors in Each Speed Bin and the Total Revenue

	Market price ($) per chip	3D Integration Strategy							
		Fast-Fast		Fast-Slow		Optimal		Random-Random	
Bin		#chips (%)	Revenues ($)	#chips (%)	Revenues ($)	#chips (%)	Revenues ($)	#chips (%)	Revenues ($)
EXTREME	1081	14	15134	3	3243	14	15134	8	8684
FAST	538	37	19906	48	25824	38	20444	35	18830
MEDIUM	325	28	9100	39	12675	36	11700	40	13000
SLOW	240	21	5040	10	2400	12	2880	17	4080
Total		100	49180 (10.28%)	100	44142 (−1.01%)	100	50158 (12.48%)	100	44594 (0.00%)

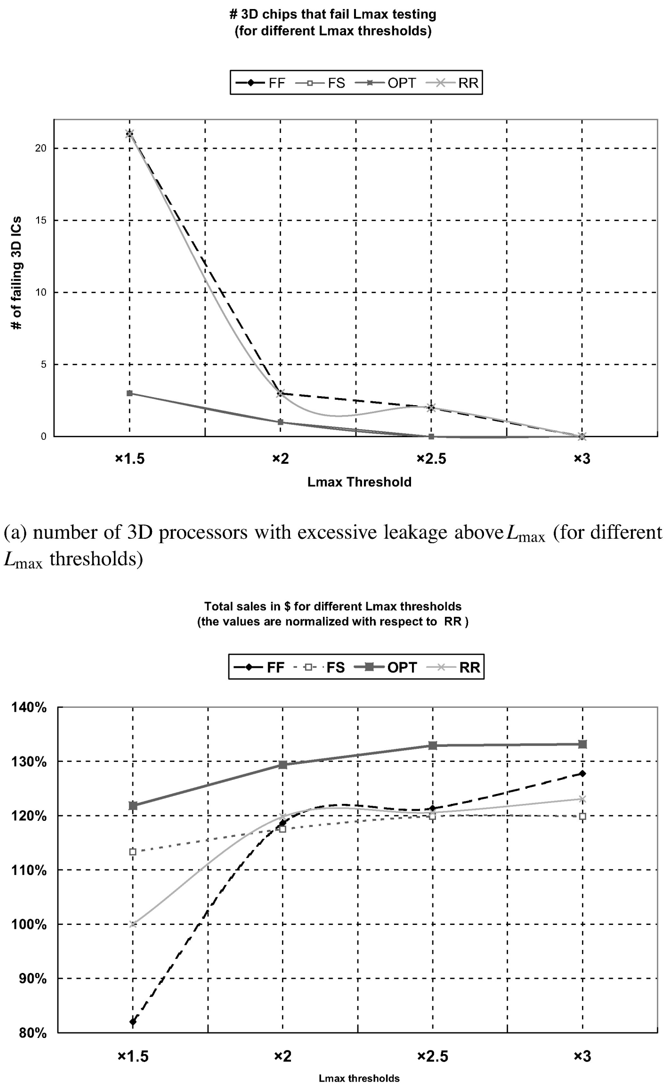
5.5 Impact on Revenue

As discussed earlier in Section 4, after fabrication and binning, IC manufacturers price according to bin. We also follow the same strategy with the 3D chips produced from our different integration strategies. With the number of processors in each bin in hand from Figure 7, we readily calculate the total revenues from applying different integration strategies and report them in Table IV. We multiply the number of processors in each bin by the market price of the bin (as given in Figure 4 according to pricegrabber.com) and sum over all bins to give the total revenues. The results show that the optimal strategy yields an increase of 12.48% in total revenues compared to the random-random strategy. The FF strategy comes second, with an increase of 10.28%.

5.6 Incorporating Leakage Current into Parametric Yield Analysis

The last stage of our experiments takes into account the leakage current during parametric yield analysis. As described earlier in Section 3, we model the total 3D IC leakage as the sum of leakage of its constituent die at the operating temperature, which is determined by the spatial and temporal variations in temperature. For our simulations, we follow a simple approach where we model the leakage consumption of the cache and CPU at a constant temperature value (i.e., at room temperature).

With a leakage-oblivious integration strategy, it is possible that a particular 3D IC assignment would exceed an imposed *leakage budget* or *threshold* ($L_{\max}$) and consequently be discarded as unusable. To illustrate this point, we choose different values for leakage threshold ($L_{\max}$) by multiplying the leakage power of the nominal 3D processor (i.e., with no variations) by 1.5, 2, 2.5, and 3. For example, an $L_{\max} = 3$ means that the leakage threshold is 3× the leakage power consumed by the nominal 3D chip at 70nm. Figure 8(a) shows the number of 3D chips that exceed the leakage threshold for the four matching strategies and for different $L_{\max}$ thresholds. As we may expect, the number of faulty chips decreases for larger values of $L_{\max}$. Furthermore, the FS and OPT strategies result in lower losses than the FF and RR strategies for smaller values of $L_{\max}$ threshold. These results agree with our discussions in Section 4. As processors with excessive leakage obviously reduce total revenues, we take this into account and compute the total revenue for different integration strategies



(a) number of 3D processors with excessive leakage above $L_{\max}$ (for different $L_{\max}$ thresholds)

(b) total revenues in for different $L_{\max}$ thresholds (values normalized with respect to RR)

Fig. 8. Impact of incorporating leakage into different 3D assignment strategies.

at different $L_{\max}$ thresholds. We plot the results in Figure 8(b). All the values reported in Figure 8(b) are normalized with respect to the RR strategy with $L_{\max} = 1.5$. At large threshold values, the OPT and FF strategies yield the largest revenue (as anticipated from Table IV). However, as $L_{\max}$ thresholds are lowered, FF loses ground as it produces systems with excessive leakage, and FS starts to look like a more promising choice. The OPT strategy holds its ground for different thresholds, as it optimally maximizes performance while discarding systems that exceed the $L_{\max}$ threshold.

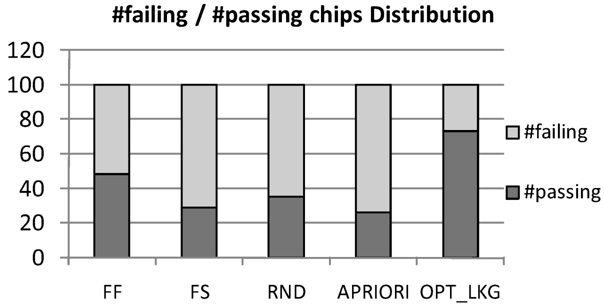


Fig. 9. Number of good and bad 3D ICs, using leakage as main binning strategy.

In another experiment, we pursue a leakage-driven strategy where binning is driven solely by leakage (as explained in Section 4.3). Here we set an aggressive leakage threshold which is equal to the sum of average CPU leakage and average L2 cache leakage. From Table I, this threshold is almost equal to 13W. Then we test the various integration strategies and count the number of chips in two leakage bins (passing and failing). Figure 9 gives the percentage of good and bad 3D ICs for the different strategies. OPT_LKG corresponds to strategy B of Section 4.3. APRIORI corresponds to a simple strategy where all CPU (and L2) chips that are below the average are thrown out in advance. Here again we see that our optimal-assignment strategy wins out over the other matching schemes. Half as many 3D ICs end up being marked as failing compared to the next-best strategy (FF). Clearly, an a priori strategy leads to an unacceptable number of failing chips.

5.7 Practical Considerations at Fabrication

While we have resorted to modeling and simulations to evaluate the impact of process variations on the CPU and L2 cache dies, the situation is actually simpler at a fabrication facility. Speed testing is executed to label the speed of each die. These speeds are then used to calculate the access time of each L2 cache die in terms of CPU cycles for each possible CPU/L2 combination. The handful of values obtained for cache latency in cycles are used as indices to a precomputed CPI lookup table (e.g., Table II). The only thing that need be computed during fabrication runtime is the assignment algorithm to figure out the optimal way to integrate the various CPUs and die. The cache-access time of the cache can be determined after fabrication. We could reasonably assume an asynchronous design for big L2/L3 caches. The Intel Montecito processor, for example, includes an asynchronous 24Mbyte L3 cache [Wuu et al. 2008]. Further, FPGA companies [Lattice 2008] have introduced programmable SRAM controllers that allow the system to select the optimum SRAM latency.

Another important consideration during fabrication is testing costs. 3D technology will only go mainstream if it is cost effective in comparison to 2D ICs. The costs of manufacturing 3D ICs depend on many factors, including the bonding technology, method of integration, and testing costs. A number of recent articles quantify the costs/benefits of using 3D ICs over 2D ICs [Smith et al. 2007a, 2007b]. From a cost presepective, the benefits from our proposed parametric yield improvement strategies should be weighed against any test and assembly costs necessary to carry them out. Smith et al. [2007b] assume the testing costs per die in 3D technology to be around $0.20. We believe that such testing must be carried out anyway, to filter out the bad die in the first place. And it is also plausible to assume that as 3D IC technology matures, the testing costs will reduce.

6. CONCLUSIONS

In this article, we have investigated the problem of maximizing parametric yield and profits in 3D integrated circuits. We have proposed strategies to model the parametric yield of 3D ICs and optimally pair different die together such that performance, leakage current, and revenues are maximized overall. We have tested our approach using a 3D processor as a potential example of 3D ICs. Compared to a strategy of randomly pairing CPUs and caches together, our optimal assignment scheme leads to an overall 6.5% improvement in MIPS and 12.5% increase in revenue. In a market where profit margins for computer systems may be relatively small, this increase in revenue can translate to a substantial increase in profits. It is also important to compare our optimal strategy to a fast-fast scheme, where the fastest CPUs are paired with the fastest caches, leaving slow processors to be paired with slow caches. While this greedy strategy may increase the total number of fastest possible 3D processors, it does so at the expense of producing a large number of slow processors and has an overall chance of producing 3D ICs with excessive leakage. In comparison, our optimal matching strategy reduces the total number of slowest processors almost in half, while maximizing the number of fastest processors that do not exceed imposed maximum leakage thresholds.

REFERENCES

Baas, B. M. 1999. A low-power, high-performance 1024-point FFT processor. *IEEE J. Solid-State Circ. 34*, 3, 380–387.

Baliga, J. 2004. Chips go vertical. *IEEE Spectrum Mag. 41*, 3, 43–47.

Banerjee, K., Souri, S. J., Kaput, P., and Saraswat, K. C. 2001. 3-D ICs: A novel chip design for deep-submicrometer interconnect performance and systems-on-chip integration. *Proc. IEEE 89*, 5, 602–633.

Benkart, P., Kaiser, A., Munding, A., Bschorr, M., Pfleiderer, H.-J., Kohn, E., Heittmann, A., Huebner, H., and Ramacher, U. 2005. 3D chip stack technology using through-chip interconnects. *IEEE Des. Test Comput. 22*, 6, 512–518.

Beyne, E. 2004. 3D interconnection and packaging: Impending reality or still a dream? In *Proceedings of the IEEE International Solid-State Circuits Conference*, 138–139.

Bhardwaj, S., Vrudhuka, S., Ghanta, P., and Cao, Y. 2006. Modeling of intra-die process variation for accurate analysis and optimization of nano-scale circuits. In *Proceedings of the Design Automation Conference*, 791–796.

Borkar, S., Karnik, T., Narendra, S., Tschanz, J., Keshavarzi, A., and De, V. 2003. Parameter variations and impact on circuits and microarchitecture. In *Proceedings of the Design Automation Conference*, 338–342.

Bowman, K., Duvall, S., and Meindl, J. 2002. Impact of die-to-die and within-die parameter fluctations on the maximum clock frequency distribution for gigascale integration. *IEEE J. Solid State Electron. 37*, 2, 183–190.

Burger, D. C. and Austin, T. M. 1997. The SimpleScalar tool set, version 2.0. Tech. Rep. CS-TR-1997-1342.

Burns, J. A., Aull, B. F., Chen, C., Chen, C.-L., Keast, C. L., Knecht, J., Suntharalingam, V., Warner, K., Wyatt, P., and Yost, D.-R. 2006. A wafer-scale 3-D circuit integration technology. *IEEE Trans. Electron. Devices 53*, 10, 2507–2516.

Cook, W. and Rohe, A. 1999. Computing minimum weight perfect matchings *INFORMS J. Comput.* 11, 38–148. http://www.or.uni-bonn.de/home/rohe/matching.html.

Cory, B. D., Kapur, R., and Underwood, B. 2003. Speed binning with path delay test in 150-nm technology. *IEEE Des. Test Comput. 20*, 5, 41–45.

Das, S., Chandrakasan, A., and Reif, R. 2004. Timing, energy, and thermal performance of three-dimensional integrated circuits. In *Proceedings of the Great Lakes Symposium on VLSI*, 338–343.

Datta, A., Bhunia, S., Choi, J. H., Mukhopadhyay, S., and Roy, K. 2006. Speed binning aware design methodology to improve profit under parameter variations. In *Proceedings of the Asia and South Pacific Design Automation Conference*, 712–717.

Davis, J. A., Venkatesan, R., Kaloyeros, A., Beylansky, M., Souri, S. J., Banerjee, K., Saraswat, K. C., Rahman, A., Reif, R., and Meindl, J. 2001. Interconnect limits on gigascale integration (GSI) in the 21st century. *Proc. IEEE 89*, 3, 305–324.

Davis, W. R., Wilson, J., Mick, S., Xu, J., Hua, H., Mineo, C., Sule, A., Steer, M., and Franzon, P. D. 2005. Demystifying 3D ICs: The pros and cons of going vertical. *IEEE Des. Test Comput. 22*, 6, 498–510.

Fukushima, T., Yamada, Y., and Koyanagi, M. 2006. New three-dimensional integration technology using chip-to-wafer bonding to acheive ultimate super-chip integration. *Japan. J. Appl. Phys. 45*, 4B, 3030–3035.

Garey, M. R. and Johnson, D. S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*, 1st (23rd printing) ed. W.H. Freeman and Company.

Grossar, E., Stucchi, M., Maes, K., and Dehaene, W. 2006. Statistically aware SRAM memroy array design. In *Proceedings of the International Symposium on Quality Electronic Design*, 25–30.

Humenay, E., Arjan, D., and Skadron, K. 2006. Impact of parameter variations on multi-core chips. In *Proceedings of the Workshop on Architectural Support for Gigascale Integration*, 1–9.

Im, S. and Banerjee, K. 2000. Full chip thermal analysis of planar (2-D) and vertically intergrated (3-D) high performance ICs. In *Proceedings of the IEEE International Electron Devices Meeting*, 727–730.

ITRS. 2008. International technology roadmap for semiconductors. `http://public.itrs.net`.

Jacob, P., Erdogan, O., Zia, A., Belemjian, P. M., Kraft, R. P., and McDonald, J. F. 2005. Predicting the performance of a 3D processor-memory chip stack. *IEEE Des. Test Comput. 22*, 6, 540–547.

Kim, C., Kim, J.-J., Chang, I.-J., and Roy, K. 2006. PVT-Aware leakage reduction for on-die caches with improved read stability. *IEEE J. Solid-State Circ. 41*, 1, 170–178.

Kuhn, H. W. 1955. The Hungarian method for the assignment problem. *Naval Res. Logist. Q. 2*, 83–97.

Lattice. 2008. LatticeMico32 asynchronous SRAM controller datasheet. `http://www.latticesemi.com/documents/doc21610x19.pdf`.

Li, F., Nicopoulos, C., Richardson, T., Xie, Y., Narayanan, V., and Kandemir, M. 2006. Design and management of 3D chip multiprocessors using network-in-memory. In *Proceedings of the International Symposium on Computer Architecture*, 130–141.

Liu, C. C., Ganusov, I., Burtscher, M., and Tiwari, S. 2005. Bridging the processor-memory performance gap with 3D IC technology. *IEEE Des. Test Comput. 22*, 6, 556–564.

Loi, G. L., Agrawal, B., Srivastava, N., L., S.-C., Sherwood, T., and Banerjee, K. 2006. A thermally-aware performance analaysis of vertically integrated (3-D) processor-memory hierarchy. In *Proceedings of the Design Automation Conference*, 991–996.

Marculescu, D. and Talpes, E. 2005. Variability and energy awareness: A microarchitecture-level perspective. In *Proceedings of the Design Automation Conference*, 11–16.

Meng, K. and Joseph, R. 2006. Process variation aware cache leakage management. In *Proceedings of the International Symposium on Low-Power Electronics*, 262–267.

Munkres, J. 1957. Algorithms for the assignment and transportation problems. *J. Soc. Industrial Appl. Math. 5*, 1, 32–38.

Orshansky, M., Milnor, L., Chen, P., Keutzer, K., and Hu, C. 2002. Impact of spatial intrachip gate length variability on the performance of high-speed digital circuits. *IEEE Trans. Comput.-Aided Des. Integr. Circ. Syst. 21*, 5, 544–553.

Patti, R. S. 2006. Three-Dimensional integrated circuits and the future of systems-on-chip designs. *Proc. IEEE 94*, 6, 1214–1224.

PTM. 2008. Predictive technology model. `http://www.eas.asu.edu/~ptm/introduction.html`.

Raj, S., Vrudhula, S. B. K., and Wang, J. 2004. A methodology to improve timing yield in the presence of process variations. In *Proceedings of the Design Automation Conference*, 448–453.

Rao, R. R., Blaauw, D., Sylvester, D., and Devgan, A. 2005. Modeling and anlysis of parametric yield under power and performance constraints. *IEEE Des. Test Comput. 22*, 4, 376–385.

Reif, R., Fan, A., Chen, K.-N., and Das, S. 2002. Fabrication technologies for three-dimensional integrated circuits. In *Proceedings of the International Symposium on Quality Electronic Design Automation*, 33–37.

Saxena, P., Menezes, N., Cocchini, P., and Kirkpatrick, D. A. 2004. Repeater scaling and its impact on CAD. *IEEE Trans. Comput.-Aided Des. Integr. Circ. Syst. 23*, 4, 451–463.

Scheiring, C. 2004. Advanced-Chip-to-Wafer technology: Enabling technology for volume production of 3D system integration on wafer level. In *Proceedings of the International Microelectronics And Packaging Society*, 1–11.

Smith, L., Smith, G., Hosali, S., and Arkalgud, S. 2007a. 3-D: It all comes down to cost. In *Proceedings of the 3-D Architectures for Semiconductor Integration and Packaging*.

Smith, L., Smith, G., Hosali, S., and Arkalgud, S. 2007b. Yield considerations in the choice of 3D technology. In *Proceedings of the IEEE International Symposium on Semiconductor Manufacturing*, 535–537.

SPEC. 2000. SPEC 2000 benchmarks. `http://www.spec.org/cpu/`.

Topol, A. W., La Tulipe, J. D. C., Shi, L., Frank, D. J., Bernstein, K., Steen, S. E., Kumar, A., Singco, G. U., Young, A. M., Guarini, K. W., and Ieong, M. 2006. Three-Dimensional integrated circuits. *IBM J. Res. Develop 50*, 4-5, 491–506.

Tsai, Y.-F., Xie, Y., Vijaykrishnan, N., and Irwin, M. J. 2005. Three-Dimensional cache design exploration using 3D cacti. In *Proceedings of the International Conference on Computer Design*, 519–524.

Wilton, S. and Jouppi, N. P. 1996. CACTI: An enhanced cache access and cycle time model. *IEEE J. Solid-State Circ. 31*, 5, 677–688.

Wuu, J., Weiss, D., Morganti, C., and Dreesen, M. 2005. The asynchronous 24MB on-chip level-3 cache for a dual-core itanium family processor. In *Proceedings of the International Solid-State Circuits Conference*, 488–612.

Xie, Y., Loh, G. H., Black, B., and Bernstein, K. 2006. Design space exploration for 3D architectures. *J. Emerg. Technol. Comput. Syst. 2,* 2, 65–103.

Zeng, A., Li, J., Rose, K., and Gutmann, R. J. 2005. First-Order performance prediction of cache memory with wafer-level 3D integration. *IEEE Des. Test Comput. 22*, 6, 548–555.

Received November 2007; revised April 2008; accepted June 2008