Scan-Chain Design and Optimization for Three-Dimensional Integrated Circuits

XIAOXIA WU and PAUL FALKENSTERN
Pennsylvania State University
KRISHNENDU CHAKRABARTY
Duke University
and
YUAN XIE
Pennsylvania State University

9

Scan chains are widely used to improve the testability of integrated circuit (IC) designs and to facilitate fault diagnosis. For traditional 2D IC design, a number of design techniques have been proposed in the literature for scan-chain routing and scan-cell partitioning. However, these techniques are not effective for three-dimensional (3D) technologies, which have recently emerged as a promising means to continue technology scaling. In this article, we propose two techniques for designing scan chains in 3D ICs, with given constraints on the number of through-silicon-vias (TSVs). The first technique is based on a genetic algorithm (GA), and it addresses the ordering of cells in a single scan chain. The second optimization technique is based on integer linear programming (ILP); it addresses single-scan-chain ordering as well as the partitioning of scan flip-flops into multiple scan chains. We compare these two methods by conducting experiments on a set of ISCAS'89 benchmark circuits. The first conclusion obtained from the results is that 3D scan-chain optimization achieves significant wire-length reduction compared to 2D counterparts. The second conclusion is that the ILP-based technique provides lower bounds on the scan-chain interconnect length for 3D ICs, and it offers considerable reduction in wire-length compared to the GA-based heuristic method.

Categories and Subject Descriptors: B.7.0 [**Integrated Circuits**]: General

General Terms: Algorithms

Additional Key Words and Phrases: 3D ICs, scan-chain design, integer linear programming, genetic algorithm, LP relaxation, randomized rounding

This research is partly supported by NSF CAREER 0643902, NSF CCF 0702617, and a grant from DARPA/IBM.

Authors' addresses: X. Wu, P. Falkenstern, and Y. Xie, Computer Science and Engineering Department, Pennsylvania State University, University Park, PA, 16802; email: {xwu,falkenst, yuanxie}@cse.psu.edu. K. Chakrabarty, Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708; email: krish@ee.duke.edu.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.
© 2009 ACM 1550-4832/2009/07-ART9 $10.00

DOI 10.1145/1543438.1543442 http://doi.acm.org/10.1145/1543438.1543442

ACM Reference Format:
Wu, X., Falkenstern, P., Chakrabarty, K., and Xie, Y. 2009. Scan-chain design and optimization for three-dimensional integrated circuits. ACM J. Emerg. Technol. Comput. Syst. 5, 2, Article 9 (July 2009), 26 pages. DOI = 10.1145/1543438.1543442 http://doi.acm.org/10.1145/1543438.1543442

1. INTRODUCTION

Scan chains are used in integrated circuit (IC) design to enhance testability and facilitate diagnosis [Makar 1998]. In a full-scan design, all flip-flops in the circuit are replaced with scan flip-flops during synthesis. These scan flip-flops are connected sequentially to form either a single scan chain or multiple scan chains. Figure 1 shows a simple example of a scan chain. Three multiplexed data flip-flops and combinational logic are shown in the figure. When the *Test* signal is low, the circuit is in normal mode (solid path), in which a flip-flop is fed by input $D1$. When the *Test* signal is high, the circuit is in test mode (dotted path) and the input to a flip-flop is provided by $D2$.

Although scan design simplifies testing and diagnosis, it suffers from area overhead due to the multiplexed-data flip-flops and the routing of stitching wires. Long scan-chain interconnects (stitching wires) increase the area of the physical layout, make routing difficult, and adversely impact performance. Since practical design-for-testability techniques must reduce the impact of test circuitry on chip performance and cost, it is important to minimize the wire-length for scan chains. Because the order of stitching scan cells does not affect the testability of the circuit, scan-cell reordering is used commonly in chip design to reduce wire-length and area overhead [Makar 1998; Hirech et al. 1998; Gupta et al. 2003].

Test application time, a major contributor to the test cost for deep-submicron ICs, is another important issue in scan design. Multiple scan chains are necessary to reduce test time for large circuits. Large scan chains are partitioned into several balanced smaller scan chains, which have almost the same scan cells and are connected to separate scan-in and scan-out pins. A key objective in scan partitioning is to minimize the longest scan-chain length for the circuit under test. In addition, since the test time is determined by the scan chain that has the largest number of scan cells among the multiple chains, multiple chain partitioning approach minimizes the test time by partitioning the scan cells into multiple balanced scan chains.

With continued technology scaling, interconnect has emerged as the dominant source of circuit delay and power consumption. The reduction of interconnect delays and power consumption are of paramount importance for deep-submicron designs. Three-dimensional (3D) ICs have recently emerged as a promising means to mitigate these interconnect-related problems [Ababei et al. 2005; Joyner and Meindl 2002; Davis et al. 2005; Xie et al. 2006]. Several 3D integration technologies have been explored recently, including wire bonded, microbump, contactless (capacitive or inductive), and through-silicon-via (TSV) vertical interconnects [Davis et al. 2005]. TSV 3D integration has the potential to offer the greatest vertical interconnect density, and therefore is the most promising one among all the vertical interconnect technologies. In 3D IC chips

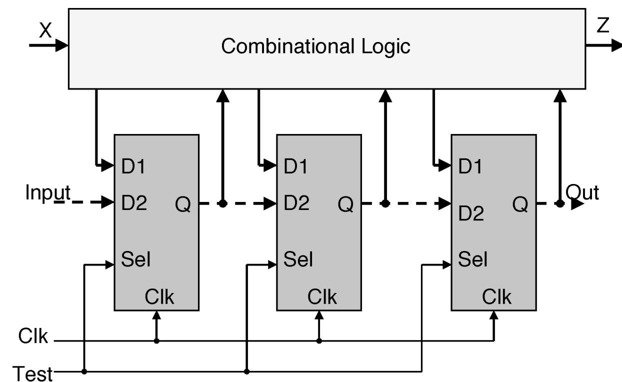

Fig. 1. Conceptual example of a scan-chain.

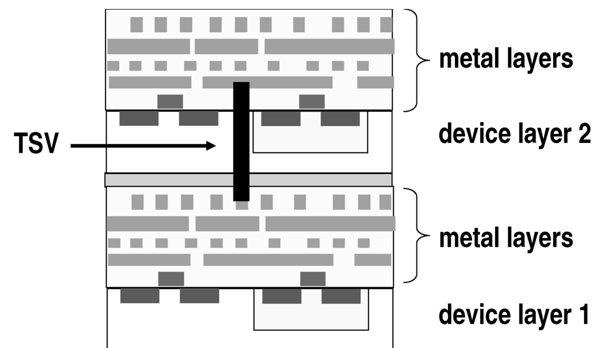

Fig. 2. A conceptual view of a 3D IC chip, with a through-silicon-via (TSV) to provide interconnect between two dies (wafers).

that are based on TSV technology, multiple active device layers are stacked together (through wafer stacking or die stacking) with direct vertical TSV interconnects [Xie et al. 2006]. Figure 2 shows the conceptual view of a 3D chip using through-silicon-via interconnects.

3D ICs offer a number of advantages over traditional two-dimensional (2D) design [Xie et al. 2006]:

—Shorter global interconnect because the vertical distance (or the length of TSVs) between two layers are usually in the range of 10 μm to 100 μm [Xie et al. 2006], depending on different manufacturers;
—Higher performance because of the reduction of average interconnect length, as well as bandwidth improvement due to die stacking;
—Lower interconnect power consumption due to wiring length reduction (reduced capacitance);
—Higher packing density and smaller footprint;
—Support implementation of mixed-technology chips: each single die can have different technologies.

The fabrication of 3D ICs is now viable. For example, in early 2007, IBM announced breakthroughs that enable the transition from horizontal 2-D chip layouts to 3-D chip stacking. Even though 3D manufacturing is becoming feasible, 3D IC design will not be commercially viable without the support of relevant 3D EDA tools and methodologies, which are needed to allow IC designers to efficiently exploit the benefits of 3D technologies.

In this article, we address the problem of minimizing the scan-chain wire-length for single scan-chain ordering, and minimizing test time and the longest wire-length among multiple scan-chains for multiple scan-chain partitioning in 3D ICs. We first propose a technique, based on a heuristic genetic algorithm (GA), for ordering scan cells in a single scan-chain. Next we formulate the relevant optimization problems and develop integer linear programming (ILP) models to address both scan-cell ordering for scan-chain routing, and scan-chain partitioning for multiple scan-chain design. Scan-chain routing is known to be equivalent to the NP-Complete Traveling Salesman Problem (TSP) [Freuer and Koo 1983], and is therefore NP-hard [Gary and Johnson 1979]. Hence, for the ILP-based method, we also investigate a heuristic technique based on a combination of LP-relaxation and randomized rounding [Raghavan and Thompson 1987]. We minimize the wire-length for single scan-chains and multiple scan-chains under constraints on the number of layers and the number of TSVs. The ILP-based optimization method also provides lower bounds on the wire-length needed for scan-chain design. Finally, we compare the results obtained using these two techniques.

The rest of the article is organized as follows: Section 2 presents related prior work on 3D ICs, scan-chain ordering, and the design of multiple scan-chains. Section 3 describes a number of techniques for designing scan-chains in 3D ICs. Section 4 presents a method based on genetic algorithms for scan-chain ordering. Section 5 describes ILP models for scan-cell ordering and partitioning, as well as the proposed technique based on LP-relaxation and randomized rounding. Section 6 presents experimental results for the ISCAS'89 benchmark circuits [Brglez et al. 1989]. Finally, Section 7 presents conclusions.

2. RELATED PRIOR WORK

Scan-chain ordering is used to reduce wire-length and power consumption [Gupta et al. 2003; Hirech et al. 1998; Huang and Huang 2006; Makar 1998]. Freuer and Koo [1983] first demonstrated that the scan-chain ordering is similar to the traveling salesman problem. Later, several algorithms were proposed to address the scan-chain ordering problem [Arora and Albicki 1987; Nakamura 1997; Lin 1996]. However, none of these previous algorithms are practical for large circuits. More recently, Makar [1998] proposed a layout-based approach, in which a scan-chain is unstitched during the scan-chain insertion process, and is reordered and connected after placement. Hirech et al. [1998] integrated scan-chain ordering into synthesis-based design reoptimization, which is carried out after floorplanning or place-and-route. Multiple scan-chains are typically used to reduce test time and decrease power consumption during test application [Ghosh et al. 2003; Il-soo et al. 2004]. In this article, the objectives

of scan-chain ordering and partitioning are to minimize wire-length and application test time.

3D technologies have attracted considerable attention in the past few years. At the fabrication level, 3D manufacturing is now viable [Reif et al. 2002; Lee et al. 2000]. Progress has also been reported on advances at the architectural level [Black et al. 2004; Kim et al. 2007]. In the 3D design automation field, several early-analysis 3D tools and 3D physical design tools have been developed in the past few years [Das et al. 2003; Cong et al. 2004; Tsai et al. 2005; Hung et al. 2006].

Among all EDA challenges for 3D IC design, tools and methodologies for 3D IC testing are regarded as the "No.1 challenge," according to a recent keynote speech [Vucurevich 2007] by Ted Vucurevich (CTO of Cadence Design System). However, research on testing for 3D ICs is still in its infancy. It is only recently that progress has been reported on the testing of 3D ICs. For example, Lewis et al. have proposed a scan-island based prebond test method to facilitate the testability of die-stacked microprocessors [Lewis and Lee 2007]. Wu et al. have proposed several 3D scan-chain design techniques [Wu et al. 2007].

In this article, a GA-based optimization technique is described first for scan-chain design. Next, ILP models are developed for scan-chain design in 3D ICs, and a combination of LP-relaxation and randomized rounding are used to derive near-optimal solutions. ILP-based scan-chain design problems are known to be NP-hard, but LP relaxation provides a lower bound such that near-optimal solutions can be quickly identified.

3. PRELIMINARIES

In this section, we describe how scan-chains can be designed for 3D ICs. First, the general 3D design flow is summarized with respect to the steps necessary for scan-chain design. Then, we present three different methods for constructing 3D scan-chains.

3.1 Design Flow for Scan-Chain Insertion

The 3D design flow, shown in Figure 3 and Figure 4, illustrates the steps involved in the design of a 3D IC and its associated scan-chains. We start with the synthesis of the target design, which leads to a gate-level netlist. Next, the scan-chain insertion tool uses the gate-level netlist to replace flip-flops with scan flip-flops. At this point, the scan-chain is unstitched. After scan-chain insertion, the logic gates are placed in the circuit. The Cadence First Encounter tool first places the circuit in a 2D space. The 3D-placement and routing tool PR3D [Das et al. 2003] partitions the 2D placement into a 3D circuit and generates the corresponding DEF (Design Exchange Format) files. In GA-based 3D flow, the ordering algorithm is performed after the 3D placement. In ILP-based flow, after 3D placement is carried out, the ordering of the scan cells and the partitioning of the scan cells into multiple scan-chains are carried out using an ILP solver. The GA algorithm and ILP solver use the cell locations and layer information from the DEF files, and the computed physical distances between the scan cells.

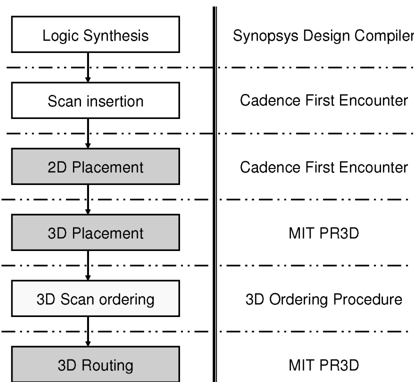

Fig. 3. GA-based 3D scan-chain design flow.

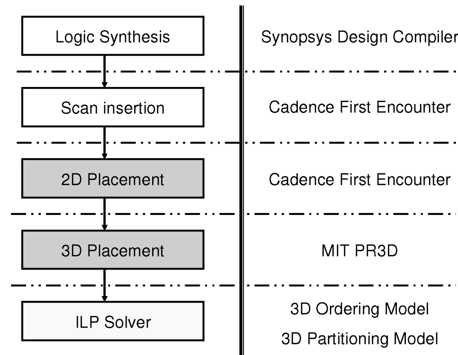

Fig. 4. ILP-based 3D scan-chain design flow.

3.2 3D Scan-Chain Design Methods

The main difference between 3D scan-chains and 2D scan-chains is the 3D placement of the scan cells. In a 2D design, the location of cell i can be represented by its XY coordinates (x_i, y_i), but in 3D design, the location must be represented by (x_i, y_i, L_i), where L_i refers to the layer where cell i is placed. As mentioned in Section 1, each wafer (die) is thinned and bonded in 3D integration. The distance between layers determines the length of a through-silicon-via (TSV), which can be in the range of $10\mu m$ to $100\mu m$ [Xie et al. 2006], depending on the 3D process and the wafer substrate type (SOI or bulk CMOS) used by various manufacturers. The TSV cross-sectional area can be as small as $1\mu m$ by $1\mu m$; however, due to the alignment constraint, the TSV pitch size is usually in the range of $5\mu m$ to $20\mu m$ [Davis et al. 2005]. The 3D locations of the scan cells and the length of TSVs should be accounted for in the 3D scan-chain construction.

A number of methods have recently been proposed for the construction of 3D scan-chains [Wu et al. 2007]. This section describes three methods using the

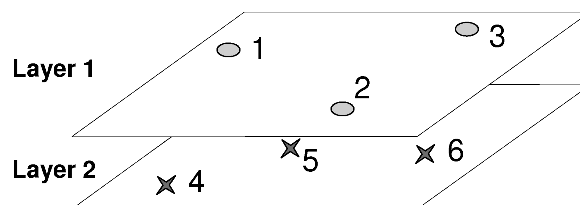

Fig. 5. A conceptual example of a 3D IC design with two layers, each of which has three scan cells to be connected.

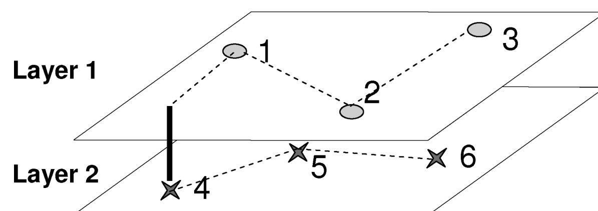

Fig. 6. Approach 1 (VIA3D): Each layer is treated independently, with a 2D scan-chain ordering method. A scan-chain in one layer is connected to a scan-chain in another layer using a a single TSV. This approach results in the minimum number of TSVs.

example 3D IC design with six scan nodes in Figure 5. The design in Figure 5 has two layers, each containing three scan nodes.

—*Approach 1 (VIA3D)*. The simplest approach to ordering a 3D scan chain is to solve the ordering problem for each layer independently using a 2D ordering algorithm. Next, all the individual scan chains are connected into one scan-chain with TSVs. If there are N layers, then there are N individual scan-chains, and $N-1$ TSVs are needed to build the final scan-chain. Figure 6 illustrates such an approach: Nodes 1, 2, and 3 are connected to form a scan-chain in layer 1; Nodes 4, 5, and 6 are connected to form a scan-chain in layer 2. A TSV (the solid line in the figure) is then used to connect these two chains to be a single chain.
  —*Advantage*: Such an approach requires no change to the scan-chain ordering algorithm: each layer is processed independently, with a 2D scan-chain ordering algorithm. The resulting number of TSVs is minimized ($N-1$ TSVs for N layers).
  —*Disadvantage*: Because it is a *locally optimized* approach, it may result in the shortest scan-chain for each layer, but the total scan-chain length may not be globally optimized.
  We call this method *VIA3D* since the number of TSVs is minimized.

—*Approach 2 (MAP3D)*. Since the vertical distance between layers is small (in the range of 10 μm to 100 μm), the second method is to transform a 3D scan-chain ordering problem into a 2D ordering problem, by mapping the nodes from several layers into a single layer (i.e., (x_i, y_i, L_i) is mapped to (x_i, y_i)). A 2D scan-chain ordering method is then applied to the design. Figure 7 illustrates such an approach. After mapping the top layer nodes (Node 1,

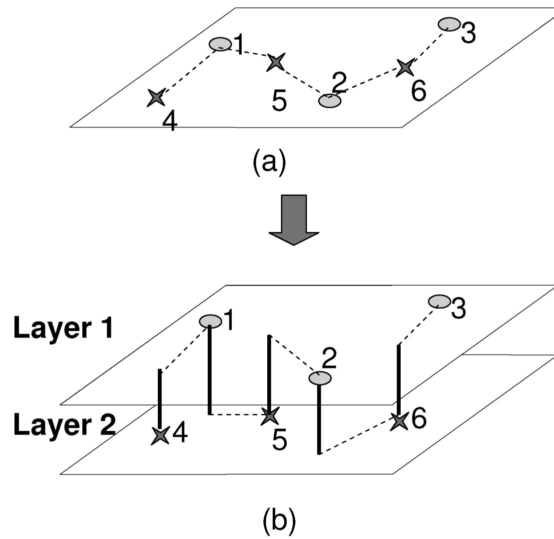

Fig. 7. Approach 2 (MAP3D): (a) All scan cells are mapped to 2D space (i.e., (x_i, y_i, L_i) is mapped to (x_i, y_i)). A 2D scan-chain ordering method is then applied to the design. (b) Such approach ignores the TSV length, and may end up to have many TSVs (the solid lines in the figure).

2, and 3) to the bottom layer, and performing 2D scan-chain ordering, the scan-chain order is 4-1-5-2-6-3. Based on such a scan-chain ordering, in 3D design, if two connected nodes are in different layers, a TSV is used. In this example, there are 5 TSVs (the solid lines in the figure).

—*Advantage*: Such an approach requires no change to the scan-chain ordering algorithm: after mapping all the nodes to a 2D plane, a 2D scan-chain ordering algorithm is applied. This is a *global* optimization method.

(a) *Disadvantage*: The vertical distance between layers is ignored. This method can therefore lead to many TSVs going back and forth between layers.

We call this method *MAP3D*, because a 3D scan-chain ordering problem is mapped to a 2D scan-chain ordering problem.

—*Approach 3 (OPT3D)*. In this method, we attempt to design a scan-chain with the least total scan-chain length, considering both horizontal and vertical wire-lengths. The distance function between two cells of this approach consists of both the horizontal Manhattan distance as well as the vertical distance between the cells. In this case, a 2D ordering algorithm cannot be used directly. The data structures (for example, the coordinates of the cells) and the code for a 2D ordering method needs to be modified to handle 3D information. Figure 8 illustrates this approach, as the scan-chain ordering is directly applied to the 3D design.

—*Advantage*: Such an approach is a true 3D scan-chain ordering optimization: The length of TSVs and the number of TSVs are considered during optimization. Users have full control over the optimization process. It is a *global* optimization method.

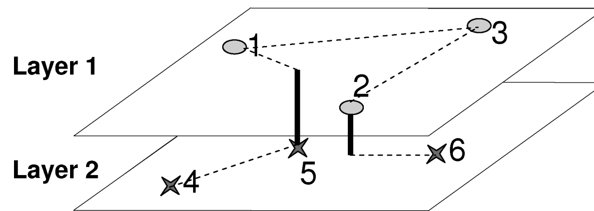

Fig. 8. Approach 2 (OPT3D): The VIA3D method is clearly a greedy approach. A true 3D scan-chain ordering method has to be developed to consider the complete design space. Such an approach takes into account the TSV length. A constraint on the number of TSVs can also be imposed.

—*Disadvantage*: Modifications to 2D scan-chain ordering algorithms are needed before they can be applied.

We call this method *OPT3D*, because it is a true 3D scan-chain ordering optimization approach.

Either of those three approaches can be used as the ordering method in the 3D scan-chain design flow. The choice of a method depends on the requirements of the design and scan-chain, such as reducing the number of TSVs, or the ease of implementation and tool integration. For example, if the goal is to minimize the total scan-chain length, OPT3D is the appropriate choice. Alternatively, the majority of the TSVs may be reserved for signal routing or thermal conduction, in which case there are tight limits on the number of TSVs, and VIA3D might be the preferred method.

4. GENETIC ALGORITHM-BASED 3D SCAN-CHAIN ORDERING

4.1 Genetic Algorithm

In this section, we develop a specific scan-chain ordering algorithm based on genetic algorithms, to evaluate different approaches proposed in Section 3. We choose a genetic algorithm based on two reasons. First, we are targeting a multiobjective optimization problem for which researchers have often used GAs by formulating the problem in terms of two-priority optimization. The primary objective here is to minimize wire-length for scan-chains. A secondary goal is to minimize test application time by using balanced scan chains. Among various randomized search algorithms, GAs and simulated annealing algorithms (SA) have been deemed in the literature to be appropriate for multiobjective optimization [Dick 2002]. A simple implementation of SA deals with only one solution at a time, and no information from previous moves is used to guide the selection of new moves [Aarts and Korst 1989]. In contrast to SA, GA maintains a pool of solutions instead of a single solution and allows communication between solutions via crossover and mutation. In this way, GA is better equipped to escape the local minima and use information from previous moves.

A genetic algorithm (GA) [Goldberg 1989] is a search and optimization method that mimics the evolutionary principles inherent in natural selection. Figure 9 illustrates the flow for the genetic algorithm. The solution is usually encoded into a string called a chromosome (in Figure 9, the chromosome is

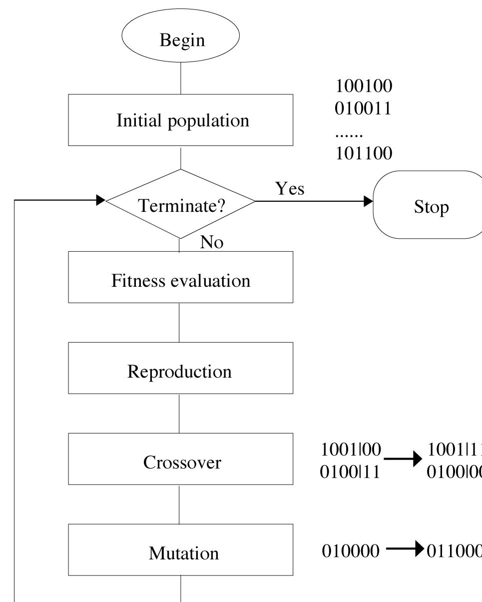

Fig. 9. Genetic algorithm flow.

encoded as binary string). Instead of working with a single solution, the search begins with a random set of chromosomes called the initial population. Each chromosome is assigned a fitness score that is directly related to the objective function of the optimization problem. The population of chromosomes is modified to a new generation by applying three operators similar to natural selection operators: reproduction, crossover, and mutation. Reproduction selects good chromosomes based on the fitness function and duplicates them. Crossover picks two chromosomes randomly and some portions of the chromosomes are exchanged with a probability P_c. Finally, the mutation operator changes a 1 to a 0 and vice versa with a small mutation probability P_m. A genetic algorithm successively applies these three operators in each generation until a termination criterion is met. It can very effectively search a large solution space while ignoring regions of the space that are not useful. In general, a genetic algorithm has the following steps: (i) generation of initial population; (ii) fitness function evaluation; (iii) selection of chromosomes; (iv) reproduction, crossover, and mutation operations.

4.2 GA-Based Scan-Chain Ordering Framework

In our GA-based scan-chain ordering framework, the solution cannot be encoded as binary string. Instead, the solution is represented by integer numbers and each flip-flop cell in the circuit is given a unique identification number. A possible solution, which is called the chromosome, is a scan-chain represented by an ordered list of numbers corresponding to the nodes, such that every node is visited exactly once. For example, if there are N scan cells in a design, integer 1 to N will be used to represent each scan cell.

The fitness function, which decides the survival chance of a chromosome (a scan-chain path), is the wire-length of this path. In the fitness evaluation stage, all the paths' fitnesses are calculated. The path with the lowest score is the path with the least wire-length and thus the best option compared to the population.

In reproduction, there is a tournament selection where the paths with a lower fitness score beat paths with higher scores. The winners of the tournament are selected to be in the next generation's population. In the crossover stage, a segment of one path is chosen and inserted in the same position into another path. However, since the second path still contains its original nodes, it contains the nodes from the segment twice (once from the original path and once from the insertion of the segment). The original position of the nodes that form the segment are deleted from the second path. This gives a legal child path that is in the next generation's population.

Instead of the classical approach to mutation, where every chromosome in the resulting population has a very small chance of mutating, in our algorithm the resulting population from reproduction and crossover is copied and mutation operates on this population. Each copied path has a probability of mutating equal to the mutation rate. The next generation's population consists of the winners of the tournament, the children of the crossover, and the result of mutation on their copies. The mutation operator in our algorithm swaps two scan cells in the path with a 25% probability and reverses the order of a segment between two scan cells in the path with a 75% probability (another mutation approach).

The fitness evaluation, reproduction, crossover, and mutation give a new population for the next generation. These steps are repeated until a set number of iterations or the termination criteria is met. The termination criteria is based on the stability of the best fitness score. If the fitness score has not improved by more than .01% over the last 1000 generations, then the algorithm is terminated.

5. OPTIMIZATION METHODS BASED ON INTEGER LINEAR PROGRAMMING

In this section, we describe the optimization methods that we use for scan-chain routing and scan-chain partitioning into multiple scan-chains. These methods are based on integer linear programming (ILP), which forms a special class of mathematical programming techniques [Bertsimas and Tsitsiklis 1997].

A mathematical programming (MP) problem involves an objective function that must be minimized (or maximized, depending on the problem formulation) under a set of constraints. If the objective function and the constraints are linear in all variables, the MP problem is referred to as a linear programming (LP) problem. In addition, if all the variables, denoted by the vector $\mathbf{x}$ are integer-valued, then it is called an integer linear programming (ILP) problem. A typical ILP model can be described as follows:

Minimize $\mathbf{Cx}$ subject to $\mathbf{Ax} \leq \mathbf{B}$, where $\mathbf{x} \geq 0$. In this model, $\mathbf{x}$ represents a vector of variables, $\mathbf{C}$ is a cost vector, $\mathbf{A}$ is a constraint matrix, and $\mathbf{B}$ refers to a vector of constants.

Objective:

Minimize $\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} \cdot w_{ij}$

Subject to:

1. $\forall i \sum_{j=1}^{n} x_{ij} = 1$
2. $\forall j \sum_{i=1}^{n} x_{ij} = 1$
3. $\sum_{i=1}^{n} x_{iu} = 0$
4. $\sum_{j=1}^{n} x_{vj} = 0$
5. $\forall i \quad x_{ii} = 0$
6. $\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ij} \cdot l_{ij} \leq L$
7. $\forall i, j \;\; u_{ij} = \sum_{k=1}^{n} x_{ik} \cdot u_{kj} + x_{ij}$
8. $\forall i, j \;\; u_{ij} + u_{ji} = 1 \; (i \neq j)$
9. $\sum_{j=1}^{n} u_{uj} = n$
10. $\sum_{i=1}^{n} u_{iv} = n$

Fig. 10. The ILP model for routing single chains.

5.1 ILP Model for Routing a Single Scan-Chain

The objective of scan-chain routing is to minimize the wire-length for the scan-chain under a set of constraints. For a 3D IC, such constraints include the number of 3D TSVs.

Let the number of scan cells being routed for the scan-chain be n. We define a binary variable x_{ij}, $(1 \leq i \leq n, 1 \leq j \leq n)$ such that $x_{ij} = 1$ if scan cell j immediately follows scan cell i in the scan-chain, and $x_{ij} = 0$ otherwise. Since the locations of the scan-in pin and scan-out pins are predefined, we choose a dummy scan cell u (source node) to be the scan-in pin and a second dummy node v (end node) to be the scan-out pin. The interconnect length from cell i to j is defined by w_{ij}, which can be obtained from the placement results. For a 3D IC, w_{ij} also accounts for the length of the TSVs, which typically ranges from 10 μm to 100 μm if cell i and j are in different layers. Due to the large pitch size of the TSVs and the need to use them for interlayer routing of data/clock/power signals and for thermal vias insertion (to reduce on-chip temperature), the number of TSVs used for scan-chain ordering is set to L as a constraint in the model. We define an array l_{ij} for modeling the layer information, which is described as follows:

$$l_{ij} = \begin{cases} 0 & \text{if } i \text{ and } j \text{ are in the same layer,} \\ 1 & \text{if } i \text{ and } j \text{ are in adjacent layer,} \\ 2 & \text{if the layer difference of } i \text{ and } j \text{ is 2,} \\ 3 & \text{if the layer difference of } i \text{ and } j \text{ is 3.} \\ \ldots & \end{cases} \tag{1}$$

The ILP model for single scan-chain routing is shown in Figure 10. The objective is to minimize the total length of the scan-chain, including scan cells u and

v. Line 1 incorporates the constraint that there is only one immediate successor per scan cell. Line 2 ensures that there is only one immediate predecessor per scan cell. Line 3 models the fact that there is no immediate predecessor for the scan-in pin, and Line 4 ensures that there is no immediate successor for the scan-out pin. Line 5 models the important non-self-loop constraint, that is, a scan cell cannot connect to itself. Line 6 models the constraint on the number of TSVs.

Lines 7–10 prevent the generation of cycles in the scan-chain. A new binary variable u_{ij} is defined, which is set to 1 if scan cell i is located before (upstream from) scan cell j in the scan-chain. Otherwise, $u_{ij} = 0$. Since the term $x_{ik} \cdot u_{kj}$ in line 7 is nonlinear, it is replaced by a new binary variable y_{ijk}. To ensure proper linearization, two new constraints are added [Iyengar and Chakrabarty 2002]:

$$x_{ik} + u_{kj} \leq y_{ijk}$$

$$x_{ik} + u_{kj} \geq 2 \cdot y_{ijk}$$

Line 8 introduces the constraint that cell i is either before cell j or after cell j in the chain. Line 9 and 10 constrain the source node and end node since the source node is before every other node and the end node is after every other node in the scan-chain.

5.2 ILP Model for the Partitioning of Scan Cells to Multiple Scan-Chains

The objective of partitioning of scan cells to multiple scan-chains is to minimize the maximum length of a scan-chain under a set of constraints. In addition, each scan-chain must be routed such that the total interconnect length is minimized. In addition, we ensure that balanced scan-chains are obtained such that the test time is minimized. In recently reported work on scan-chain design for 3D ICs [Wu et al. 2007], only scan-chain routing for single scan-chains was considered; the more practical problem of multiple scan-chains was not addressed.

Assume that there are a total of n scan cells in a circuit. Let the number of scan-chains be m. We define a binary variable x_{ijk}, $1 \leq i \leq n, 1 \leq j \leq n$, such that $x_{ijk} = 1$ if scan cell j immediately follows scan cell i in scan-chain k. Each scan-chain k has dummy scan-in cell u_k and dummy scan-out cell v_k. The interconnect length from cell i to cell j is defined by w_{ij}. As before, we set the limit on the number of TSVs to be L and define the data structure array l_{ij} to incorporate layer-related information.

The model for multiple chain partitioning is shown in Figure 11. In this formulation, the objective of multiple chain partitioning formulation is to minimize the maximum length of all the multiple chains. Line 1 provides a constraint that there is only one immediate successor per scan cell in each scan-chain. Line 2 ensures that there is only one immediate predecessor per scan cell in each scan-chain. Line 3 implies that there is no immediate predecessor for scan-in pin in each scan-chain. Line 4 ensures that there is no immediate successor for scan-out pin in each scan-chain. Line 5 models the limitation that one scan cell cannot connect to itself. Lines 6 and 7 indicate that one cell can only be in one scan-chain. Line 8 gives number limitation of through silicon vias. Line 9

Objective:

Minimize $\max_{k=1\ to\ m}(\sum_{i=1}^{n}\sum_{j=1}^{n} x_{ijk} \cdot w_{ij})$

Subject to:

1. $\forall i,k\ \ \sum_{j=1}^{n}\sum_{k=1}^{m} x_{ijk} = 1$
2. $\forall j,k\ \ \sum_{i=1}^{n}\sum_{k=1}^{m} x_{ijk} = 1$
3. $\forall j\ \ \sum_{i=1}^{n}\sum_{k=1}^{m} x_{iu_jk} = 0$
4. $\forall i\ \ \sum_{j=1}^{n}\sum_{k=1}^{m} x_{v_ijk} = 0$
5. $\forall i,k\ \ x_{iik} = 0$
6. $\forall i\ \ \sum_{j=1}^{n}\sum_{k=1}^{m} x_{ijk} = 1$
7. $\forall j\ \ \sum_{i=1}^{n}\sum_{k=1}^{m} x_{ijk} = 1$
8. $\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{m} x_{ijk} \cdot l_{ij} \leq L$
9. $\forall k,\ \ \lfloor n/m \rfloor \leq \sum_{i=1}^{n}\sum_{j=1}^{n} x_{ijk} + 1 \leq \lceil n/m \rceil$
10. $\forall i,j,k\ \ u_{ijk} = \sum_{h=1}^{n} x_{ihk} \cdot u_{hjk} + x_{ijk}$
11. $\forall i,j,k\ \ u_{ijk} + u_{jik} \leq 1$
12. $\sum_{j=1}^{n}\sum_{k=1}^{m} u_{u_kj} = n$
13. $\sum_{i=1}^{n}\sum_{k=1}^{m} u_{iv_k} = n$

Fig. 11. The ILP model for partitioning scan cells into multiple scan-chains and for scan-chain routing.

guarantees balanced scan cells in each chain. We constraint the number of scan cells in each chain to be in a range shown in the figure. Therefore, each chain has almost the same number of scan cells, thus reducing the test time. Lines 10-13 prevent the generation of cycles in each scan-chain. As in Figure 10, a new binary variable u_{ijk} is defined, indicating whether cell i is located before cell j in the ordered scan-chain. Since the term $x_{ihk} \cdot u_{hjk}$ in line 10 is nonlinear, it is replaced by a new binary variable y_{ijhk}. In addition, two new constraints are added:

$$x_{ihk} + u_{hjk} \leq y_{ijhk}$$

$$x_{ihk} + u_{hjk} \geq 2 \cdot y_{ijhk}.$$

Line 11 brings a constraint that cell i either is before cell j or after cell j in the scan-chain if those two cells are in that chain. Lines 12 and 13 constrain the source nodes and end nodes of each chain since the source node is before every other node and the end node is after every other node in each chain.

5.3 Randomized Rounding

While the ILP models discussed in this section can be used to optimally solve the scan-chain routing and partitioning problems, they do not scale well for large circuits. ILP is known to be an NP-hard problem [Gary and Johnson 1979]; however, LP problems can be solved optimally in polynomial time [Bertsimas and Tsitsiklis 1997]. Therefore, we adopt the method of LP relaxation and combine

it with randomized rounding [Raghavan and Thompson 1987]. In LP relaxation, the binary variables are relaxed to real-valued variables such that the solution to the relaxed LP problem provides a lower bound on the total wire-length for the scan-chain(s). However, the fractional values obtained for the x_{ij} variables are inadmissible in practice; these variables must be mapped to either 0 or 1. For this purpose, we use the method of randomized rounding.

The randomized rounding technique for ILP problems consists of three steps. The first step is to solve the corresponding LP problem, fixing all x_{ij} variables that are assigned to 1. The second step is to randomly pick a variable from the set of variables with fractional values and assign it to 1 with a probability equal to the fractional value. For example, if the solution to the LP problem assigns the value 0.4 to a variable, we generate a random number between 0 and 1. If this random number is less than or equal to 0.4, the variable is set to 1. Otherwise it is set to be 0. In the third step, the LP problem is solved again, and the randomized rounding step is repeated until all variables are set to either 0 or 1.

Note that in the Step 2 of randomized rounding technique, the violation of constraints must be prevented in order to ensure that the LP problem is feasible. For example, in single scan-chain design, we randomly pick a variable x_{ij} with a fractional value, and assign the value 1 to it according to the random number that we generate. Before the assignment, we check if x_{ik} $(j \neq k)$ or x_{kj} $(i \neq k)$ is already assigned to 1. If this is the case, we can only assign 0 to x_{ij}, otherwise a constraint (line 1 or line 2 of the ILP model) is violated, and the LP problem becomes infeasible. Similarly, in multiple scan-chain partitioning, before we assign 1 to x_{ijk}, we need to guarantee there is no constraint violation with existing fixed-variable values. For example, if x_{ijh} is already fixed to 1 then x_{ijk} $(k \neq h)$, x_{kjh} $(k \neq i)$, and x_{ikh} $(k \neq j)$ cannot be assigned to 1.

6. EXPERIMENTS AND RESULTS

6.1 GA-Based Results

To evaluate our genetic algorithm based 3D scan-chain ordering approaches, we implemented the 3D scan-chain ordering algorithm and conducted experiments on a set of ISCAS89 benchmark circuits [Brglez et al. 1989]. All experiments were performed on a dual Intel Xeon processor (3.2GHz, 4GB RAM) Linux machine. In this experiment, we use MIT Lincoln Lab's 180nm 3D library [Davis et al. 2005] to perform the synthesis and placement.

In Table I, we summarize the wire-length comparison among 2D scan-chain ordering, VIA3D, MAP3D, and OPT3D approaches under GA-based design. The distance between two layers is set to be $10\mu m$. The first column gives the circuit names selected from the ISCAS89 benchmarks and the number of flip-flop cells (included in the bracket). The number of the flip-flop cells ranges from 74 in $s1423$ to 1728 in $s35932$. The second column provides the wire-length result obtained from 2D scan-chain ordering which is also based on genetic algorithm symmetric TSP. The third column is the layer number in 3D circuits, ranging from 2 to 4 layers. The fourth to ninth columns show the wire-length and

Table I. The Comparison Between Various GA-Based Ordering Methods for a Single Scan-Chain

Circuits (No. of Flip-Flops)	2D Wire Length (μm)	No. of Layers	Wire-Length for VIA3D (μm)	TSV Number for VIA3D	Wire-Length for MAP3D (μm)	TSV Number for MAP3D	Wire-Length for OPT3D (μm)	TSV Number for OPT3D	Reduction: VIA3D Over 2D	Reduction: MAP3D Over 2D	Reduction: OPT3D Over 2D
$s1423$ (74)	2319	2	1776	1	1510	34	1545	22	23.4%	34.9%	33.3%
		3	1523	2	1361	56	1122	22	34.3%	41.3%	51.6%
		4	1429	3	1296	74	1009	30	38.3%	44.1%	56.4%
$s5378$ (179)	5004	2	4156	1	3477	100	3163	48	16.9%	30.5%	36.8%
		3	3263	2	3173	148	2572	50	34.8%	36.6%	48.6%
		4	3107	3	3411	220	2549	56	37.9%	31.8%	49.1%
$s9234$ (211)	6241	2	4684	1	4253	106	3868	48	24.9%	31.8%	38.0%
		3	3578	2	3571	160	3067	56	42.7%	42.8%	50.8%
		4	3576	3	3897	252	2771	70	42.7%	37.5%	55.6%
$s15850$ (534)	14913	2	10832	1	10562	236	9424	128	27.3%	29.1%	36.8%
		3	8864	2	9133	432	7944	146	40.5%	38.8%	46.7%
		4	7852	3	9456	600	7025	176	47.3%	36.6%	52.9%
$s13207$ (638)	16500	2	11907	1	11742	364	11188	144	27.8%	28.8%	32.2%
		3	9946	2	11704	616	9284	180	39.7%	29.0%	43.7%
		4	8667	3	11613	794	8014	194	47.4%	29.6%	51.4%
$s35932$ (1728)	61969	2	40913	1	38674	926	36846	462	33.9%	37.5%	40.5%
		3	31834	2	33747	1436	30880	590	48.6%	45.5%	50.1%
		4	26327	3	35295	2074	28057	660	57.5%	43.0%	54.7%
average									37.0%	36.1%	46.0%

Table II. GA OPT3D Wire-Length Results With a Limit on the Number of TSVs.

Circuits	No. of layers	Wire-length (with limit on the number of TSVs)	Number of TSVs (with limit on the number of TSVs)	Wire-length (without limit on the number of TSVs)	Number of TSVs (without limit on the number of TSVs)
$s1423$	2	1559	18	1545	22
(74)	3	1125	18	1122	22
	4	1037	20	1009	30
$s5378$	2	3545	20	3163	48
(179)	3	2921	20	2572	50
	4	2735	20	2549	56
$s9234$	2	4312	20	3868	48
(211)	3	3285	20	3067	56
	4	2976	20	2771	70
$s15850$	2	9733	100	9424	128
(534)	3	7982	100	7944	146
	4	7463	100	7025	176
$s13207$	2	11405	100	11188	144
(638)	3	9905	100	9284	180
	4	8031	100	8014	194
$s35932$	2	43524	100	36846	462
(1728)	3	33490	100	30880	590
	4	32488	100	28057	660

via number resulting from VIA3D, MAP3D, and OPT3D, in which the unit of wire-length is μm. The last three columns provide the wire length reduction of VIA3D, MAP3D, and OPT3D approaches over 2D ordering.

From the table, one can observe that OPT3D can achieve the best wire length reduction for the scan-chain design. The average reduction from 2D to OPT3D is 46.0% and the maximum reduction is 56.4%. The average reduction from 2D to VIA3D and MAP3D are 37.0% and 36.1%, respectively.

When the number of layers increases, often the scan-chain length of MAP3D increases, contrary to the expected results. This happens for MAP3D between layers 3 and 4 for most of the circuits. The increase of the scan-chain length for MAP3D can be attributed to the large via count in the scan-chain. The distance traveled between the layers of the scan-chain accumulates with a large via count, thus increasing the scan-chain length.

Table II gives the wire-length result with TSV number limits in the OPT3D approach. In GA-based approach, the limit of TSVs is taken in account by fitness function. We add an extra constraint (the number of TSVs) in the fitness function. If the number of TSVs exceed the predefined number then the result from the fitness function is set to be infinite so that this path will not considered to be a new population. The limit on the number of TSVs is set to 20 for the smallest three circuits and 100 for the three largest circuits. It shows that limiting the number of TSVs increases the wire-length but provides a means to control the routing congestion caused by vias. It also indicates that users can have control of the optimization process according to the different requirements.

6.2 ILP-Based Results and Comparison

In ILP-based design, we used Xpress-MP,[1] a commercial ILP solver, to solve the ILP models for single scan-chain ordering, and for multiple scan-chain partitioning and ordering. We conducted the experiments on the same ISCAS'89 benchmark circuits as GA-based design. We evaluate VIA3D and OPT3D in ILP-based approach because these two methods are more reasonable and practical than MAP3D method.

6.2.1 *Single Scan-Chain Ordering.* Table III presents results on the scan-chain interconnect length for single scan-chain routing. We examine the following methods in terms of their effectiveness: I2D (ILP 2D), IVIA3D (ILP VIA3D), ILP3D (OPT3D without TSV limit), and ILP3DV (OPT3D with TSV limit). The first column gives the names of the circuits selected from the ISCAS'89 benchmarks and the number of flip-flops (in parenthesis) in these circuits. The second column provides the wire-length (the unit is μm) results obtained from 2D scan-chain ordering. The third column refers to the number of layers in 3D circuits, and it is varied from 2 to 4. The fourth to ninth columns show the wire-length and the number of TSVs resulting from IVIA3D, ILP3D, and ILP3DV. In ILP3DV, the limit on the number of TSVs is set to 20 for the three smallest circuits and 100 for the three largest circuits. The last three columns provide the wire-length reduction provided by these three approaches compared to 2D ordering.

We draw several conclusions from the results in Table III. First, all the 3D approaches lead to lower wire-length than 2D ordering, which indicates an inherent advantage of using 3D ICs. The average reduction from I2D to ILP3D is up to 55.4%, and on average it is 45.5%. The reductions in wire-length from I2D to IVIA3D and ILP3DV are 37.1% and 41.8%, respectively. Second, the wire-length is reduced as the number of 3D layers increases. However, the benefit of wire-length reduction, when we move from 3 layers to 4 layers, is smaller than the case when we move from 2 layers to 3 layers, which implies that the wire-length-related benefit of 3D technology does not increase with an increase in the number of layers. On the other hand, with an increase in the number of layers, the die-stacking cost increases and thermal challenges become more pronounced due to higher power density. Therefore, it is important to balance performance benefits with cost/thermal trade-offs. This observation is consistent with the conclusions drawn from previous work [Vaidyanathan et al. 2007; Xie et al. 2006], where it was reported that the maximum number of layers in a 3D IC should be no more than 4–5. The third observation is that ILP3D and ILP3DV provide better improvements over a 2D approach than IVIA3D. The reason is that ILP3D and ILP3DV are true 3D optimization solutions rather than based on an existing 2D approach. The fourth observation is that the wire-length becomes longer with tighter limits on the number of TSVs; this can be attributed due to a corresponding reduction in the availability of potentially shorter vertical interconnects provided by the TSVs. However, limiting the number of TSVs can provide a means to control the routing congestion

[1] www.dashoptimization.com.

Table III. The Comparison between Various ILP-Based Ordering Methods for a Single Scan-Chain

Circuits (No. of Flip-Flops)	I2D Wire Length (μm)	No. of Layers	Wire-Length for IVIA3D (μm)	TSV Limit for IVIA3D	Wire-Length for ILP3D (μm)	TSV Limit for ILP3D	Wire-Length for ILP3DV (μm)	TSV Limit for ILP3DV	Reduction: IVIA3D Over I2D	Reduction: ILP3D Over I2D	Reduction: ILP3DV Over I2D
$s1423$ (74)	2245	2	1726	1	1441	22	1493	20	23.1%	35.8%	33.5%
		3	1450	2	1065	22	1075	20	35.4%	52.6%	52.1%
		4	1375	3	1000	16	1023	20	34.3%	55.4%	54.4%
$s5378$ (179)	4951	2	4028	1	3118	58	3375	20	18.6%	37.0%	31.8%
		3	3032	2	2534	50	2725	20	38.6%	48.8%	44.9%
		4	2996	3	2387	60	2567	20	39.4%	51.8%	48.1%
$s9234$ (211)	6060	2	4408	1	3635	48	3854	20	27.2%	40.0%	36.4%
		3	3277	2	2851	56	3022	20	45.9%	52.9%	50.1%
		4	3140	3	2698	66	2872	20	48.1%	55.4%	52.6%
$s15850$ (534)	14022	2	9566	1	9144	126	9354	100	31.7%	34.7%	33.3%
		3	8379	2	7658	152	7870	100	40.2%	45.3%	43.8%
		4	7485	3	6703	154	7004	100	46.6%	52.2%	50.0%
$s13207$ (638)	15340	2	11822	1	10476	132	11075	100	29.4%	31.7%	27.8%
		3	9813	2	8752	146	9575	100	41.2%	42.9%	37.6%
		4	8311	3	7646	150	7953	100	46.4%	50.1%	48.1%
$s35932$ (1728)	51265	2	38040	1	34351	450	37167	100	25.8%	32.9%	27.5%
		3	30270	2	26118	520	31209	100	40.9%	49.0%	39.1%
		4	25768	3	25549	576	29032	100	49.7%	50.1%	43.3%
Average									37.1%	45.5%	41.8%

caused by TSVs and the utilization of TSVs for various other purposes such as data/clock signal routing and power routing. Therefore, designers can make appropriate trade-offs between wire-length and number of TSVs, based on different requirements.

Table IV provides a comparison of wire-length between the GA-based technique from Wu et al. [2007] and the proposed ILP-based approaches. As in the case of the ILP-based methods, the GA-based methods are referred to as follows: G2D (GA 2D), GVIA3D (GA VIA3D), GA3D (GA OPT3D without a limit on the number of TSVs), and GA3DV (GA OPT3D with a limit on the number of TSVs). The ILP-based results are not reproduced in Table IV due to lack of space. The reader is referred to Table III. In the second column, the LP lower bound for I2D is given. In Columns 9–11, we list the LP lower bounds for IVIA3D, ILP3D, and ILP3DV, respectively. We can see from Table III and Table IV that the difference between LP relaxation before randomized rounding technique (LP lower bound) and the final results after randomized rounding is within 5%, thereby highlighting the efficiency of the randomized rounding technique, as well as demonstrating that LP-relaxation provides a tight lower bound. Another observation is that ILP-based methods can always achieve shorter wire-length than GA methods. The wire-length given by I2D is 6.2% less on average than that for G2D. On average, IVIA3D provides a 6.4% reduction in wire-length compared to GVIA3D. ILP3D and ILP3DV achieve 5.3% and 5.9% reduction over GA3D and GA3DV, respectively. Even though the combination of LP-relaxation and randomized rounding does not guarantee optimal solutions, it provides near-optimal results. Table V provides the CPU run-time comparison (in seconds) between the GA and ILP methods. The results show that ILP methods always run faster than GA methods.

In Table III, the limit on the number of TSVs is fixed and the comparison between wire-length with and without TSVs limit is given. Here, we examine the impact of TSVs limit on the wire length and select $s15850$ and $s13207$ to conduct the experiments. Figure 12 and Figure 13 show the wire-length for the scan-chain in $s15850$ and $s13207$ under different limits on the number of TSVs, which ranges from 20 to 100. It shows that the wire-length decreases with an increase in the number of TSVs, but this decrease is significant only for four layers. For two layers and three layers, the number of TSVs has relatively less impact on the scan-chain wire-length. Therefore, the number of TSVs for scan-chains can be set based on the number of layers, as well as wire-length and die area considerations.

6.2.2 *Partitioning and Routing for Multiple Scan-Chains.* In this section, we present the results obtained using the ILP-based methods for multiple scan-chains. Table VI shows ILP-based 2D and OPT3D results with two chains and four chains, respectively. The chain length showed in the table represents the longest chain length among balanced multiple scan-chains. We list 2D wire-length with single, two, and four chains in Columns 2–4. Column 6–11 provide the wire-length and number of vias for ILP3D with one, two, and four scan-chains. The results indicate that with more chains, the maximum routing length among multiple chains is reduced. Since each chain has almost the same

Table IV. Comparison between GA-Based and ILP-Based Methods for Scan-Chain Ordering

Circuits (No. of Flip-Flops)	Lower Bound I2D	Wire Length G2D	Reduction: I2D Over GA	No. of Layers	Wire Length GVIA3D	Wire Length GA3D	Wire Length GA3DV	Lower Bound IVIA3D	Lower Bound ILP3D	Lower Bound ILP3DV	Reduction: IVIA3D Over GA	Reduction: ILP3D Over GA	Reduction: ILP3DV Over GA
*s*1423	2197	2319	3.1%	2	1776	1545	1559	1719	1405	1465	2.8%	6.7%	4.2%
(74)				3	1523	1122	1125	1423	1030	1063	4.7%	5.1%	4.4%
				4	1429	1009	1037	1351	973	1012	3.7%	0.9%	1.4%
*s*5378	4825	5004	1.0%	2	4156	3163	3545	3915	3045	3302	3.1%	1.4%	4.8%
(179)				3	3263	2572	2921	2918	2460	2659	7.1%	1.5%	6.7%
				4	3107	2549	2735	2894	2309	2508	3.5%	6.4%	6.1%
*s*9234	5954	6241	3%	2	4684	3868	4312	4347	3547	3835	5.8%	6.0%	10.6%
(211)				3	3578	3067	3285	3222	2769	3007	8.4%	7.0%	8.0%
				4	3576	2771	2976	3095	2611	2872	12.2%	2.6%	3.5%
*s*15850	13752	14913	6.0%	2	10832	9424	9733	9401	8891	9146	11.6%	3.0%	3.9%
(534)				3	8864	7944	7982	8225	7408	7768	5.4%	3.6%	1.4%
				4	7852	7025	7463	7356	6517	6882	4.6%	4.6%	6.1%
*s*13207	15004	16500	7.0%	2	11907	11188	11405	11605	10019	10749	9.1%	6.4%	2.9%
(638)				3	9946	9284	9905	9604	8307	9446	9.4%	5.7%	3.3%
				4	8667	8014	8031	8137	7646	7953	5.3%	4.6%	1.0%
*s*35932	49353	61969	17.3%	2	40913	36846	43524	36841	32540	35684	7.1%	6.8%	14.6%
(1728)				3	31834	30880	35490	29135	24409	29940	4.9%	15.4%	12.0%
				4	26327	28057	32488	24803	23987	27833	2.2%	8.9%	10.6%
Average			6.2%								6.4%	5.3%	5.9%

Table V. CPU Run-Time (seconds) Comparison between GA and ILP Methods

Circuits	G2D	I2D	No.of layers	GVIA3D	IVIA3D	GA3D	ILP3D
$s1423$	8.2	4.6	2	22.6	2.1	7.1	1.9
			3	22.7	1.6	7.6	1.6
			4	21.0	1.2	7.9	0.9
$s5378$	43.9	25.9	2	80.7	5.7	46.2	16.3
			3	119.4	6.5	47.9	30.7
			4	69.9	3.8	35.2	21.2
$s9234$	56.1	41.4	2	96.9	6.0	63.2	46.5
			3	143.0	9.4	46.2	31.6
			4	110.8	7.5	77.6	48.3
$s15850$	977.4	155	2	616.8	61	1011	131.3
			3	890.2	32	782.5	116.1
			4	771.4	45	938.9	101.2
$s13207$	1552	314.8	2	1550	51	1366	138.6
			3	1571	48.4	1319	204.6
			4	1064	28.7	1495	175
$s35932$	21003	8141	2	16254	1383	22733	7491
			3	18763	594	19474	5356
			4	8652	259.4	19380	3268

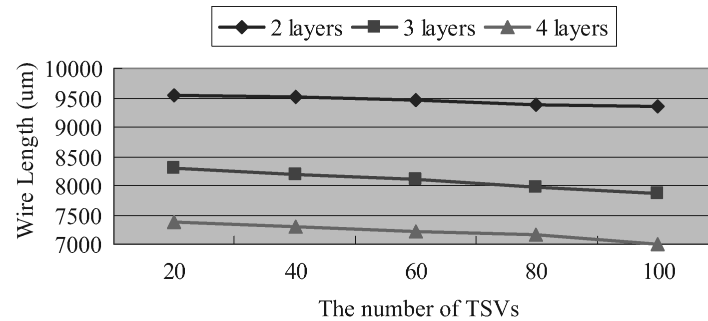

Fig. 12. The impact of the number of TSVs on scan-chain wire-length for s15850.

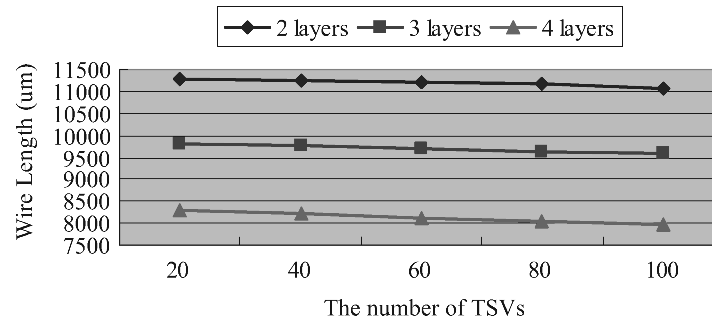

Fig. 13. The impact of the number of TSVs on scan-chain wire-length for s13207.

Table VI. Results for ILP-Based Multiple Scan-Chain Partitioning and Routing

	I2D wire-length				ILP3D wire-length			ILP3D TSV limit		
Circuits	1 chain	2 chains	4 chains	No. of layers	1 chain	2 chains	4 chains	1 chain	2 chains	4 chains
s1423	2245	1717	1387	2	1441	1097	743	22	26	21
				3	1065	823	553	22	26	25
				4	1000	774	577	16	28	35
s5378	4951	3875	2652	2	3118	2343	1351	58	56	52
				3	2534	1979	1172	50	64	74
				4	2387	1729	1092	60	66	90
s9234	6060	4508	2846	2	3635	2959	1763	48	58	56
				3	2851	2245	1444	56	54	59
				4	2698	1912	1332	66	84	96
s15850	14022	10407	6909	2	9144	6585	3903	126	128	125
				3	7658	5205	2981	152	164	171
				4	6703	4773	2763	154	156	176
s13207	15340	11484	7413	2	10476	7279	4328	132	162	164
				3	8752	5939	3564	146	166	185
				4	7646	4863	3018	150	160	192

number of scan cells, the test time is also reduced compared to single scan-chain design.

6.2.3 *Multiple Clock Domains and Multiple Voltage Islands.* The previous results are based on a single clock domain and a single voltage island. If we consider multiple clock domains or multiple voltage islands, one can see that there are some similarities between 2D multiple clock domains/multiple voltage islands and 3D design [Makar 1998; Lackey et al. 2002]. Each clock domain/voltage island can be considered a "layer" and the clock/voltage domain crossings need to be minimized. However, in multiple clock domains, all the scan cells in the same clock domain are grouped together [Makar 1998], which is more restrictive than 3D scan-chain routing. To minimize clock skew during shift, all scan-chains should be ordered such that all flops in the same clock domain are grouped together. The number of lockup latches, inserted between the clock domains, is therefore fixed [Illman and Aldrich 2002]. In multiple voltage islands, voltage level shifter is needed to realize the voltage change between different voltage islands [Lackey et al. 2002]. In addition, scan-chain design for multiple voltage islands also raises the power-sequencing issue. One solution is that power-sequencing circuitry is held to the power-on state during test operation. Another solution is that each power-sequenced island is tested independently [Lackey et al. 2002]. Therefore, the combination of multiple clock domains/multiple voltage islands and 3D design needs special consideration and will be considered as part of future work.

7. CONCLUSIONS

Scan-chain design is an important design-for-testability problem for emerging 3D ICs. We have shown that scan-design routing techniques for 2D ICs leads to unnecessarily long wire-lengths for 3D chips. We have proposed GA-based scan-chain ordering and ILP-based scan-chain ordering and partitioning. In ILP modeling, a combination of LP-relaxation and randomized rounding is also proposed. We have described several optimization techniques for designing scan-chains in 3D ICs, with given constraints on the number of through-silicon-vias (TSVs). We have compared the ILP-based optimization methods to a heuristic technique based on the use of a genetic algorithm (GA). We have obtained tight lower bounds on the scan-chain interconnect length for 3D ICs using ILP-based optimizations, and demonstrated that considerable reduction in wire-length is achieved over 2D design techniques and over the GA-based heuristic method.

ACKNOWLEDGMENTS

The authors gratefully acknowledge IBM 3D Program Managers Kerry Bernstein and Albert Young for their invaluable help with the understanding of the 3D fabrication process.

REFERENCES

AARTS, E. AND KORST, J. 1989. *Simulated Annealing and Boltzmann Machines*. Wiley, Chichester, U.K.

ABABEI, C., FENG, Y., GOPLEN, B., MOGAL, H., ZHANG, T., BAZARGAN, K., AND SAPATNEKAR, S. S. 2005. Placement and routing in 3D integrated circuits. *IEEE Des. Test Comput. 22*, 6, 520–531.

ARORA, R. AND ALBICKI, A. 1987. Computer-aided scan path design for self-testing chips. In *Proceedings of Midwest Symposium on Circuits and Systems*. 301–304.

BERTSIMAS, D. AND TSITSIKLIS, J. 1997. *Introduction to Linear Optimization*. Athena Scientific.

BLACK, B., NELSON, D. W., WEBB, C., AND SAMRA, N. 2004. 3D processing technology and its impact on iA32 micro-processors. In *Proceedings of the IEEE International Conference on Computer Design*. 316–318.

BRGLEZ, F., BRYAN, D., AND KOZMINSKI, K. 1989. Combinational profiles of sequential benchmark-circuits. In *Proceedings of the IEEE International Symposium on Circuits and Systems*. vol. 3. 1929–1934.

CONG, J., WEI, J., AND ZHANG, Y. 2004. A thermal-driven floorplanning algorithm for 3D ICs. In *Proceedings of the International Conference on Computer-Aided Design*. 306–313.

DAS, S., CHANDRAKASAN, A., AND REIF, R. 2003. Design tools for 3-D integrated circuits. In *Proceedings of the Asia and South Pacific Design Automation Conference*. 53–56.

DAVIS, W. R., WILSON, J., MICK, S., XU, J., HUA, H., MINEO, C., SULE, A. M., 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.

DICK, R. P. 2002. Multi-objective synthesis of low-power real-time distributed embedded systems. Ph.D. thesis, Princeton University.

FREUER, M. AND KOO, C. 1983. Method for rechaining shift register latches which contain more than one physical book. *IBM Tech. Disclo. Bull. 25*. 4818–4820.

GARY, M. R. AND JOHNSON, D. S. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman.

GHOSH, D., BHUNIA, S., AND ROY, K. 2003. Multiple scan-chain design technique for power reduction during test application in BIST. In *Proceedings of the International Symposium on Defect and Fault Tolerance in VLSI Systems*. 191–198.

GOLDBERG, D. 1989. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley, New York.

GUPTA, P., KAHNG, A. B., AND MANTIK, S. 2003. Routing-aware scan-chain ordering. In *Proceedings of the Asia and South Pacific Design Automation Conference*. 857–862.

HIRECH, M., BEAUSANG, J., AND XINLI, G. 1998. A new approach to scan-chain reordering using physical design information. In *Proceedings of the International Test Conference*. 348–355.

HUANG, X. L. AND HUANG, J. I. 2006. A routability constrained scan-chain ordering technique for test power reduction. In *Proceedings of the Asia and South Pacific Conference on Design Automation*. 5 pp.

HUNG, W. L., LINK, G. M., XIE, Y., NARAYANAN, V., AND IRWIN, M. J. 2006. Interconnect and thermal-aware floor-planning for 3D micro-processors. In *Proceedings of the International Symposium on Quality Electronic Design*.

IL-SOO, L., YONG MIN, H., AND AMBLER, T. 2004. The efficient multiple scan-chain architecture reducing power dissipation and test time. In *Proceedings of the Asian Test Symposium*. 94–97.

ILLMAN, R. AND ALDRICH, G. 2002. On the finish rests with multiple clock. http://www.us.design-reuse.com/articles/2820/on-time-finish-rests-with-multiple-clocks.html.

IYENGAR, V. AND CHAKRABARTY, K. 2002. Test bus sizing for system-on-a-chip. *IEEE Trans. Comput. 51,* 5, 449–459. 0018-9340.

JOYNER, J. W. AND MEINDL, J. D. 2002. Opportunities for reduced power dissipation using three-dimensional integration. In *Proceedings of the Interconnect Technology Conference*. 148–150.

KIM, J., NICOPOULOS, C., PARK, D., DAS, R., XIE, Y., VIJAYKRISHNAN, N., AND DAS, C. 2007. A novel dimensionally-decomposed router for on-chip communication in 3D architectures. In *Proceedings of the International Symposium on Computer Architecture*.

LACKEY, D. E., ZUCHOWSKI, P. S., BEDNAR, T. R., STOUT, D. W., GOULD, S. W., AND COHN, J. M. 2002. Managing power and performance for system-on-chip designs using voltage islands. In *Proceedings of the International Conference on Computer-Aided Design*. 195–202.

LEE, K. W., NAKAMURA, T., ONO, T., YAMADA, Y., MIZUKUSA, T., HASHIMOTO, H., PARK K. T., KURINO, H., AND KOYANAGI, M. 2000. Three-dimensional shared memory fabricated using wafer stacking technology. In *Proceedings of the International Electron Devices Meeting*. 165–168.

LEWIS, D. L. AND LEE, H. S. 2007. A scan-island based design enabling pre-bond testability in die-stacked micro-processors. In *Proceedings of the International Test Conference.*

LIN, K. 1996. Layout-driven chaining of scan flip-flops. In *IEE Prcedings-Comput. Digit. Tech.* vol. 143.

MAKAR, S. 1998. A layout-based approach for ordering scan-chain flip-flops. In *International Test Conference*. 341–347.

NAKAMURA, K. 1997. Scan paths wire length minimization and its short path error correction. In *NEC Res. Devel*. *38*, 22–27.

RAGHAVAN, P. AND THOMPSON, C. 1987. Randomized rounding: A technique for provably good algorithms and algorithmic proofs. *Combinatorica 7,* 4, 365–374.

REIF, R., FAN, A., CHEN, K., AND DAS, S. 2002. Fabrication technologies for three-dimensional integrated circuits. In *Proceedings of the International Symposium on Quality Electronic Design*. 33–37.

TSAI, Y., XIE, Y., NARAYANAN, V., AND IRWIN, M. J. 2005. Three-dimensional cache design exploration using 3D cacti. In *Proceedings of the International Conference on Computer Design*. 519–524.

VAIDYANATHAN, B., HUNG, W., WANG, F., XIE, Y., NARAYANAN, V., AND IRWIN, M. J. 2007. Architecting micro-processor components in 3D design space. In *Proceedings of the VLSI Design*. 103–108.

VUCUREVICH, T. 2007. The long road to 3D integration: Are we there yet? In *Keynote Speech at the 3D Architecture Conference*.

WU, X., FALKENSTERN, P., AND XIE, Y. 2007. Scan-chain design for three-dimensional (3D) ICs. In *Proceedings of the International Conference on Computer Design*.

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.

Received January 2008; revised July 2008; accepted July 2008 by Paul Franzon