Fast Scan-Chain Ordering for 3-D-IC Designs Under Through-Silicon-Via (TSV) Constraints

Christina C.-H. Liao, Allen W.-T. Chen,
Louis Y.-Z. Lin, and Charles H.-P. Wen

Abstract—This brief addresses the problem of scan-chain ordering under a limited number of through-silicon vias (TSVs), and proposes a fast two-stage algorithm to compute a final order of scan flip-flops. To enable 3-D optimization, a greedy algorithm, multiple fragment heuristic, is modified and combined with a dynamic closest-pair data structure, FastPair, to derive a good initial solution in stage one. Stage two initiates two local refinement techniques, 3-D planarization and 3-D relaxation, to reduce the wire (and/or power) cost and to relax the number of TSVs in use to meet TSV constraints, respectively. Experimental results show that the proposed algorithm results in comparable performance (in terms of wire cost only, power cost only, and both wire-and-power cost) to a genetic-algorithm method but runs two-order faster, which makes it practical for TSV-constrained scan-chain ordering for 3-D-IC designs.

Index Terms—3-D IC, scan testing, through-silicon via (TSV).

I. INTRODUCTION

3-D integration provides many advantages over the traditional 2-D implementation, such as better packaging efficiency and higher transistor density [1]–[4]. Among all vertical-integration techniques, through-silicon via (TSV) provides the best timing and power performance for interconnection. However, TSVs typically incur additional area overhead and may become another source of defects [5]. Therefore, considering yield loss and area cost, the number of TSVs in use is typically limited in a 3-D IC design.

Scan design is the most prevailing design-for-testability (DFT) technique, which aims to reduce the difficulty of testing on the circuit-under-test (CUT). To further study interconnection on 3-D-IC designs, Yuan *et al.* [6] showed that the scan-stitching wire length in a multilayer circuit is shorter than that in a planar circuit. Experimental results in [6] also suggested that the more TSVs in use in the scan chain, the less scan-stitching wire cost. Such observation combined with the TSV-induced yield loss indicates a tradeoff between the scan-stitching wire and the number of TSVs in use. Therefore, a constraint of TSVs must be included for 3-D-IC designs.

Moreover, both pre-bond testing and post-bond testing are thriving research topics for 3-D ICs. For pre-bond testing and post-bond testing, a 3-D DFT architecture with inter-die TSV-based interconnects was proposed in [7]. Marinissen *et al.* [8] presented an optimization of 3-D DFT containing TSVs for pre-bond testing. For enabling pre-bond testability,

Lewis *et al.* [9] proposed a scan-island-based design, and Kumar *et al.* [10] proposed a hyper-graph-based partitioning for pre-bond 3-D-IC testing. Chen *et al.* [11] used on-chip TSV monitoring to test TSV before bonding. Additionally, several scan-ordering approaches based on genetic algorithm (GA) for 3-D-IC post-bond testing were accordingly proposed in [6], where its runtime issue remains unresolved and the solution quality remains unstable. Moreover, scan-induced power dissipation has not yet been considered previously. Therefore, we are motivated to develop a fast 3-D scan-chain design for post-bond testing, which can optimize wire and power costs simultaneously.

In this brief, TSV-constrained scan-chain ordering is first analyzed and formulated into a traveling salesman problem (TSP). Later, a fast algorithm is developed to minimize the scan-stitching wire length (and/or scan-induced power dissipation), and to simultaneously satisfy the constraint on the number of TSVs in use for 3-D-IC designs. Our algorithm consists of two stages. First, we construct an initial simple path through all scan flip-flops (FFs) using a modified greedy algorithm, multiple fragment heuristic, via a dynamic closest pair data structure, FastPair. Second, we propose two new techniques, 3-D planarization and 3-D relaxation, to minimize the wire (and/or power) cost and to reduce TSV usage, respectively. Experimental results demonstrate the practicality of our algorithm by producing comparable scan-stitching wire length and/or total power dissipation to the GA method with a two-order speedup on average.

As a result, the contributions of this brief can be summarized as: 1) formulating the scan-chain ordering with the consideration of TSV constraints into a modified TSP problem and 2) proposing a fast algorithm for scan-chain ordering of 3-D-IC designs to simultaneously minimize costs and meet TSV constraints. The rest of this brief is organized as follows. In Section II, we present problem formulations of TSV-constrained scan-chain ordering with three different objectives (minimizing wire, power, and both wire-and-power cost). In Section III, a fast two-stage algorithm proposed for scan-chain ordering is detailed. Section IV presents the experimental results, which include various comparison between our algorithm and a GA method. Finally, we draw our conclusions in Section V.

II. PROBLEM FORMULATION OF SCAN-CHAIN ORDERING FOR TSV-CONSTRAINED 3-D-IC DESIGNS

In this section, we formulate the scan-chain ordering problem for 3-D-IC designs with three different objectives: 1) to minimize the scan-stitching wire length to avoid routing congestion and timing violation; 2) to reduce the scan-induced power dissipation on testing to avoid damage and reliability degradation to the CUT; and 3) to simultaneously optimize wire and power costs. Defined below are two necessary notations used in the remainder of this brief.

1) $V^k \triangleq \{v_0, v_1, \ldots, v_{n-1}\}$: n-bit input pattern k where v_i is scanned in scan FF c_i during scan testing.

2) $R^k \triangleq \{r_0, r_1, \ldots, r_{n-1}\}$: n-bit output response k where r_i is scanned out from scan FF c_i during scan testing.

The problem of the scan-chain ordering to minimize the wire-and-power (combined) cost can be formulated into:

Input: CUT C with n scan FFs $\{c_0, c_1, \ldots, c_{n-1}\}$, their layer information $\{L_0, L_1, \ldots, L_{n-1}\}$, a fixed set of m test patterns $\{V^1, R^1, V^2, R^2, \ldots, V^m, R^m\}$; and a user-defined coefficient α used to adjust the weight of wire and power costs.

Output: A scan-FF order is formed and denoted by $\langle c'_0, c'_1, \ldots, c'_{n-1}\rangle$ such that the combined cost $[(1 - \alpha)\times$ wire_cost $+ \alpha\times$ power_cost$]$ is minimized under TSV constraints

where the wire_cost is defined in [6], the power_cost is defined in [12] and user-defined coefficient α ranges from 0 to 1. When $\alpha = 0$, this problem becomes a wire-cost minimization problem; when $\alpha = 1$, only power cost is considered, as in the power-cost minimization problem.

III. Fast Two-Stage Scan-Chain Ordering

In this section, the proposed algorithm is elaborated with respect to different objectives, including wire-cost minimization in Section III-A, power-cost minimization in Section III-B, and wire-and-power (combined) cost minimization in Section III-C, respectively.

A. Minimizing Wire Cost

Regarding the formulation for the wire minimization concerning TSV-based 3-D-IC designs, two approaches are proposed in [6]. One approach is developed on the basis of GA, and the other is based on integer linear programming (ILP). Although the GA approach may possibly find a near-optimal solution, the quality of one identified solution cannot be guaranteed. Moreover, the ILP approach, which can find the optimal cost, may not be able to produce a feasible solution within a limited time.

Considering the practicality, our algorithm needs to be developed to overcome the runtime issue. Fig. 1 shows the overall flow. According to Fig. 1, in stage 1, a latest tour-construction heuristic used in TSP problems, multiple fragment heuristic [13], is incorporated. Moreover, a dynamic closest-pair data structure, FastPair [14], is implemented to facilitate the considerable computation of pair-wise cost in the tour-construction heuristic. An initial solution is obtained in stage 1 and sent to stage 2 to perform 3-D planarization to optimize the total cost, and 3-D relaxation to reduce the total number of TSVs in use.

1) Initial Order Computation: First, a good initial ordering of scan FFs needs to be constructed in stage 1. We solve this problem using the multiple fragment heuristic [13] combined with a dynamic closest-pair data structure, FastPair [14]. This heuristic finds the shortest edge between the endpoints of two different paths until all points are connected. Each point is initialized as a one-point path. Then, the next min-cost edge can be quickly found by the closest-pair data structure. Meanwhile, the useless point will be deleted from the point set.
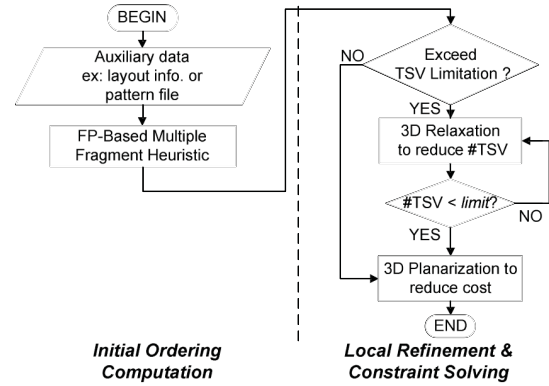

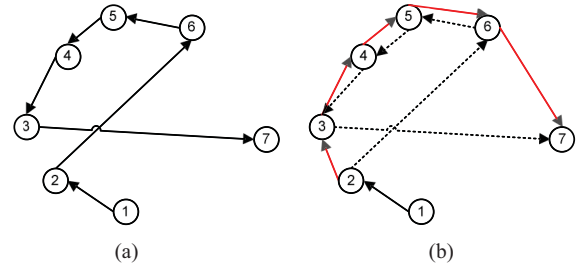Fig. 1. Proposed flow for fast scan-chain ordering.

Fig. 2. Planarization for a six-point example. (a) Initial solution. (b) After planarization.

Finally, this procedure iterates until a simple path is derived where all points are stitched.

Before exploring FastPair, we first outline the concept of the neighbor heuristic where each point p stores its nearest point from the point set S based on the following equation:

$$d(p) = \min_{q \in S - \{p\}} D(p, q) \quad (1)$$

where $D(p, q)$ is a user-defined function and computes the distance between scan FFs. That is

$$D(p, q) = |x_p - x_q| + |y_p - y_q| + c_{\text{TSV}} \times |L_p - L_q|. \quad (2)$$

FastPair [14] is developed on the basis of the neighbor heuristic with more improvements. It runs in the time complexity of $O(n \log^2 n)$ and outperforms the neighbor heuristic empirically [14]. Considering the time complexity, FastPair and its operations are incorporated when developing our multiple-fragment-heuristic-based algorithm.

2) Local Refinement and Constraint Solving: After obtaining the initial solution, the second stage of our algorithm applies two strategies to optimize total wire cost and/or to relax TSVs in use. Fig. 2(a) shows an initial path with the unoptimized wire cost. In the study of the optimization theory, 2-D planarization is one common technique to reduce the total cost in the TSP problem. The key idea is to planarize a graph and remove all cross edges on the plane. Fig. 2(b) shows such an example. Cross edges, (2, 6) and (3, 7) are replaced by (2, 3) and (6, 7).

From a different point of view, such an operation can be viewed as the reverse of a fragment of one path, i.e., the sequence of node traversal. Reversing the fragment from

$6 \rightarrow 5 \rightarrow 4 \rightarrow 3$ into $3 \rightarrow 4 \rightarrow 5 \rightarrow 6$ in the original path can effectively reduce the total wire cost. To further generalize this idea, we reverse any possible fragment with the edge length from 1 to $(n-1)$ and test if the reversion can reduce cost successfully. Such local-refinement technique is termed 3-D planarization and runs in the time complexity of $O(n \log^2 n)$. More details are provided in [14].

Fig. 3 shows an example of using 3-D planarization to reduce the total wire cost. In Fig. 3, $L1$, $L2$, $L3$, and $L4$ represent the first, second, third, and fourth layers, respectively, and the connection of two scan FFs residing in two consecutive layers requires one TSV. Fig. 3 shows an example with refinements of both the wire cost and the total number of TSVs in use. The left part of Fig. 3 represents an initial scan-chain order: $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5$. After reversing the fragment $2 \rightarrow 3 \rightarrow 4$ to $4 \rightarrow 3 \rightarrow 2$, shown as the right part in Fig. 3, the total wire cost is reduced, meanwhile saving two TSVs.

A similar constraint-solving technique, 3-D relaxation, is proposed to reduce the total number of TSVs in use from all layers for meeting the TSV constraints. 3-D relaxation reverses all fragments of 1 to $(n-1)$ edges again to find the best reduction of TSVs in use until the target TSV number is achieved. Later, 3-D planarization is also performed to further reduce the total wire cost but uses no extra TSVs. The time complexity for the constraint solving technique is also $O(n \log^2 n)$. To sum up, the proposed scan-chain ordering is more efficient than the previous work [6] in terms of time, meanwhile considering the TSV constraints simultaneously.

Table I shows the preliminary performance of our algorithm with the TSV constraints on four sample circuits. The second column denotes the number of partitions. The third column shows the TSV constraints. The fourth column (under no. of TSV) reports the TSV usage in the initial order followed by the TSV usage in the wire-cost minimized order. The fifth and sixth columns under WC show the total wire cost in μm after performing 3-D relaxation and 3-D planarization, respectively. The seventh column reports the reduction rate from constraint solving to local refinement in stage 2. Note that the total wire cost after stage 1 is not presented since the initial solution is not legal due to an overuse of TSVs. That is, it is meaningless to compare such illegal wire cost with two legal solutions under WC. As a result, Table I shows that all TSV constraints can be satisfied and good reduction can be achieved. Particularly, the four-layer s38417 has the best reduction rate of 65.79% after constraint solving. For these four circuits, the 3-D planarization technique achieves averagely a 44.38% reduction in the total wire cost.

B. Minimizing Power Cost

For the power minimization concerning TSV-based 3-D-IC designs, Giri *et al*. from [12] have developed a GA approach. However, this time-consuming and unstable approach, may impair quality solutions. Therefore, we propose a new flow, as illustrated in Fig. 1, to deal with this power-optimization problem. At the beginning, we establish a look-up table storing
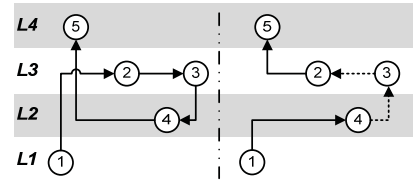

Fig. 3. 3-D planarization for reducing wire cost and two TSVs.

TABLE I
3-D RELAXATION AND 3-D PLANARIZATION ON
WIRE AND TSV REDUCTION

Circuit (no. of FF)	no. of L	No. of TSV Const.	No. of TSV Init./final	WC a. Relax.	WC b. Planar.	red.(%): (a−b)/a
s9234 (211)	3	20	71/19	8916	5088	**42.93**
	4	20	81/19	8698	5428	**37.59**
	5	20	50/20	5044	4078	**19.15**
s15850 (597)	3	100	181/99	17314	12764	**26.28**
	4	100	239/99	24606	12778	**48.07**
	5	100	153/99	14774	10820	**26.76**
s38417 (1636)	3	200	445/199	58294	33166	**43.11**
	4	200	592/200	100906	34516	**65.79**
	5	200	536/200	79730	32972	**58.65**
s35932 (1728)	3	200	628/200	72208	34826	**51.77**
	4	200	505/199	70540	32370	**54.11**
	5	200	552/200	79288	33002	**58.38**
Average						**44.38**

the pair-wise cost to avoid repeated calculations. Since the objective involves the transitions of positions in the scan chain, there are several modifications in the proposed algorithm.

According to the problem formulation, the pattern information is an input to the algorithm, and the objective is to minimize the total weighted transitions (TWT) (refer to [12]), which can be viewed as $\text{TWT}(V, R) = \text{VWT}(V) + \text{RWT}(R) + \text{PWT}(V, R)$. $\text{VWT}(V)$ and $\text{RWT}(R)$ denote the weighted transitions for the input vector V and the output response R, respectively. $\text{PWT}(V, R)$ denotes the weighted peak transition. The computation for the TWTs requires the knowledge of an initial scan-chain order and the position information. In the proposed flow, the scan-induced transitions between scan FFs are available at the beginning.

For computing an initial solution, we change the user-defined function $D(p, q)$ in (1) to reflect the scan-induced transitions between scan FFs under m test patterns. That is

$$D(p, q) = \sum_{j=1}^{m} \left[\left(v_p^j \oplus v_q^j \right) + \left(r_p^j \oplus r_q^j \right) \right] \qquad (3)$$

where the notations have the same definition as those in Section II. Since each computation of (3) costs $O(m)$ for m test patterns, a look-up table that stores the pair-wise transitions is established to avoid redundant calculations in the proposed algorithm. After constructing a scan-chain order, the sum of the total transitions between scan FFs is minimized by multiple fragment heuristic. Furthermore, we improve the TWTs by rotating it n times and choose the best solution.

Table II shows the performance of our algorithm with TSV constraints. The fifth and sixth columns show the TWTs (under

TABLE II
3-D RELAXATION AND 3-D PLANARIZATION ON
POWER AND TSV REDUCTION

Circuit	No. of	No. of TSV		TWT		red.(%):
(no. of FF)	L	Const.	Init./final	a. Relax.	b. Planar.	(a−b)/a
s9234 (211)	3	20	200/20	2.26E+06	2.02E+06	**10.62**
	4	20	296/20	2.3E+061	2.02E+06	**12.56**
	5	20	390/20	2.36E+06	2.04E+06	**13.56**
s15850 (597)	3	100	358/100	1.46E+07	1.34E+07	**8.22**
	4	100	546/100	1.49E+07	1.37E+07	**8.05**
	5	100	872/100	1.51E+07	1.37E+07	**9.27**
s38417 (1636)	3	200	1400/200	9.28E+07	8.28E+07	**10.78**
	4	200	1920/200	9.62E+07	8.35E+07	**13.20**
	5	200	2562/200	9.64E+07	8.40E+07	**12.86**
s35932 (1728)	3	200	1430/200	1.40E+07	1.01E+07	**27.86**
	4	200	2230/200	1.53E+07	1.04E+07	**32.03**
	5	200	2818/200	1.76E+07	1.10E+07	**37.50**
Average						**16.38**

TWT) after performing 3-D relaxation and 3-D planarization, respectively. Other columns in this table have the same meanings as in Table I. Table II also shows good reduction in the total number of TSVs in use on the big circuits, especially on s35932. Particularly, the five-layer s35932 has the best reduction rate of 37.50% after 3-D relaxation. Consequently, 3-D planarization achieves averagely 16.38% in total power-cost reduction.

C. Minimizing Wire-and-Power Cost Simultaneously

The same flow illustrated in Fig. 1 can be used for the combined-cost optimization problem. According to the problem formulation, the inputs to the algorithm include test patterns and layout information, and the objective is to optimize the combined cost (both the wire and power costs). For computing an initial solution, we combine the user-defined function $D(p, q)$ in (2) and (3) to reflect the combined cost between scan FFs under m test patterns

$$D(p, q) = (1 - \alpha) \times D_{\text{wire}}(p, q) + \alpha \times D_{\text{power}}(p, q) \quad (4)$$

where $D_{\text{wire}}(p, q)$ and $D_{\text{power}}(p, q)$ are derived from (2) and (3); coefficient α ranges from 0 to 1.

IV. EXPERIMENTAL RESULTS

A reference flow for 3-D scan designs can be found in [6]. We modify the flow and utilize various commercial softwares to complete 3-D scan designs. We first partition an original design into N layers which minimizes the number of TSVs in use and balances the area between different layers [15]. Different partitioning strategies result in different total wire/power costs. However, our scan-chain ordering algorithm can effectively minimize cost under different scenarios. After obtaining N-layer designs, synopsys design compiler does logic synthesis. Finally, all FFs and TSVs can be placed in N-layer designs through a 3-D TSV-aware placement [16], which particularly considers the positions of TSV insertion. On the basis of such placement result, our wire-cost model can be further applied.

We also perform scan insertion on the original design and then retrieve the test information via Standard Test Interface Language (STIL) files using Synopsys TetraMax. Two input data, including the layout information and test patterns, are necessary for reducing the scan-induced power in the proposed algorithm. Using the same input data, we also perform the proposed algorithm to minimize wire and power costs simultaneously.

For fair comparison, the settings in the previous GA approach [6] are used in our experiments. The population size is set to 2000 and the same operators are used, which include reproduction, crossover, and mutation. The GA stops when no more than 0.0001% improvement on the fitness score (i.e., the total scan-stitching wire length or TWTs) can be obtained for the last 1000 generations.

Both the proposed 3-D scan-chain ordering algorithm and the previous GA approach are exercised on a Linux machine with a Pentium Core Duo (2.4 GHz) processor and 4-GB memory. TSMC 0.18-μm library is used and the height of a TSV is set as 10 μm, whereas the partitions for 3-D-ICs range from 3 to 5. ISCAS'89 benchmark circuits are used to conduct experiments. Experiments are divided into three parts, minimizing wire cost, minimizing power cost, and minimizing wire and power cost simultaneously.

A. Minimizing Wire/Power Cost

Table III (and Table IV) reports the results of comparing the wire cost (and power cost) between a GA method and our algorithm under different TSV constraints for all benchmark circuits. The second column specifies the maximum number of TSVs that can be used. Note that the benchmark circuits of different scales are imposed with different TSV constraints, from 20 to 200. The third column shows the number of partitions for layers. The fourth and fifth columns show the total wire cost in μm (and TWTs) after performing the GA method and performing the proposed algorithm, respectively. The sixth column shows the improvement ratio (denoted by over.) of our approach compared with the GA method. The last column shows the speed-ups (denoted by sdup) of our algorithm compared with the GA method. The runtime of the proposed algorithm is proportional to the number of iterations used to perform 3-D planarization and circuit size. Although, the proposed algorithm results in the slightly inferior wire cost on s15850, it produces comparable (or slightly better) wire cost (and power cost) than the GA method on two other larger circuits. Moreover, the proposed algorithm can run at least two-orders faster than the GA method.

B. Minimizing Wire-and-Power Costs Simultaneously

We further perform the proposed algorithm under TSV constraints while simultaneously minimizing wire and power costs. Table V reports the results of comparing the combined costs between the GA method and our algorithm, under TSV constraints 20 for circuit s5378 (the total number of FFs is 179) with three-layer partitions. The first column specifies the coefficient α in (4) with a stepping rate 0.2 from 0 to 1. The second, third, and fourth columns show the comparison

TABLE III
WIRE COST AND RUNTIME COMPARISONS

Circuit (no. of FF)	TSV const.	TWTs no. of L	WC		over. (%)	sdup (X)
			GA	our		
s9234 (211)	20	3	5310	5088	−4.18	922
		4	5182	5428	4.75	725
		5	4166	4078	−2.11	1228
s15850 (597)	100	3	12 636	12 764	1.01	514
		4	12 180	12778	4.91	415
		5	11 094	10 820	−2.47	1067
s38417 (1636)	200	3	34 566	33 166	−4.05	393
		4	34 752	34 516	−0.68	226
		5	33 744	32 972	−2.29	337
s35932 (1728)	200	3	35 748	34 826	−2.58	287
		4	33 864	32 370	−4.41	433
		5	33 418	33 002	−1.24	403
Average					−1.11	579

TABLE IV
POWER COST AND RUNTIME COMPARISONS

Circuit (no. of FF)	TSV const.	No. of L	TWT		over. (%)	sdup (X)
			GA	our		
s9234 (211)	20	3	1.99E+06	2.02E+06	1.51	282
		4	2.02E+06	2.02E+06	0.00	195
		5	2.03E+06	2.04E+06	0.49	172
s15850 (597)	100	3	1.36E+07	1.34E+07	−1.47	407
		4	1.38E+07	1.37E+07	−0.72	317
		5	1.36E+07	1.37E+07	0.74	317
s38417 (1636)	200	3	8.86E+07	8.28E+07	−6.55	230
		4	8.76E+07	8.35E+07	−4.68	188
		5	8.94E+07	8.40E+07	−6.04	153
s35932 (1728)	200	3	1.05E+07	1.01E+07	−3.81	364
		4	1.08E+07	1.04E+07	−3.70	268
		5	1.11E+07	1.10E+07	−0.90	211
Average					−2.04	258

TABLE V
WIRE-AND-POWER AND RUNTIME COMPARISONS ON s5378

α	WC		over. (%)	TWT		over. (%)	sdup (X)
	GA	our		GA	our		
0	3618	3602	0.00	925720	933772	0.01	381
0.2	3628	3600	−0.01	908644	927748	0.02	274
0.4	3764	3508	−0.07	893832	907906	0.02	268
0.6	4022	3862	−0.04	876088	882418	0.01	299
0.8	5904	6088	0.03	824076	821180	0.00	449
1	19 418	20 888	0.08	747609	744027	0.00	208
Average			−0.01	Average		0.01	313

of total wire cost in μm between the GA method and the proposed algorithm. The fifth, sixth, and seventh columns also compare the total power cost between two methods. The last column shows the speed-ups (denoted by sdup) of our algorithm compared with the GA method. According to Table V, as α increases, power cost decreases, but wire cost increases. The coefficient α can be adjusted by users at their discretion. Again, the proposed algorithm can achieve comparable or better performance in terms of wire and/or power cost with at least two-order runtime speedups to the GA method.

V. CONCLUSION

A fast scan-chain ordering algorithm for 3-D-IC designs considering TSV constraints was developed, consisting of two stages: 1) initial order computation and 2) local refinement and constraint solving. We convert the stage-1 problem into a TSP problem, and use multiple fragment heuristic combined with FastPair to quickly derive a good initial solution. Two techniques, 3-D planarization and 3-D relaxation were also proposed to minimize the cost (wire, power or combined ones) and to relax the total number of TSVs in use, respectively. Experimental results showed that the proposed algorithm can achieve comparable or better performance than a GA method, with at least two-order runtime speedups when considering TSV constraints. As a result, the proposed algorithm can be practically used for scan-chain ordering of 3-D-IC designs.

REFERENCES

[1] J. W. Joyner, P. Zarkesh-Ha, J. A. Davis, and J. D. Meindl, "A three-dimensional stochastic wire-length distribution for variable separation of strata," in *Proc. IEEE Int. Interconn. Technol. Conf.*, Burlingame, CA, Jun. 2000, pp. 126–128.

[2] C. Ababei, Y. Feng, B. Goplen, H. Mogal, T. Zhang, K. Bazargan, and S. Sapatnekar, "Placement and routing in 3D integrated circuits," *IEEE Design Test. Comput.*, vol. 22, no. 6, pp. 520–531, Nov.–Dec. 2005.

[3] W. R. Davis, J. Wilson, S. Mick, J. Xu, H. Hua, C. Mineo, A. M. Sule, M. Steer, and P. D. Franzon, "Demystifying 3D ICs: The pros and cons of going vertical," *IEEE Design Test. Comput.*, vol. 22, no. 6, pp. 498–510, Nov.–Dec. 2005.

[4] Y. Xie and Y. Ma, "Design space exploration for 3D integrated circuits," in *Proc. 9th Int. Conf. Solid-State Integr.-Circuit Technol.*, Beijing, China, 2008, pp. 2317–2320.

[5] B. Noia, K. Chakrabarty, and E. J. Marinissen, "Optimization methods for post-bond die-internal/external testing in 3D stacked ICs," in *Proc. IEEE Int. Test Conf.*, Austin, TX, Nov. 2010, pp. 1–9.

[6] X. Wu, P. Falkenstern, K. Chakrabarty, and Y. Xie, "Scan-chain design and optimization for three-dimensional integrated circuits," *ACM J. Emerg. Technol. Comput. Syst.*, vol. 5, no. 2, pp. 1–26, Jul. 2009.

[7] J. Li and D. Xiang, "DFT optimization for pre-bond testing of 3D-SICs containing TSVs," in *Proc. IEEE Int. Conf. Comput. Design*, Amsterdam, The Netherlands, Oct. 2010, pp. 474–479.

[8] E. J. Marinissen, C.-C. Chi, J. Verbree, and M. Konijnenburg, "3D DFT architecture for pre-bond and post-bond testing," in *Proc. IEEE Int. 3D Syst. Integr. Conf.*, Munich, Germany, Nov. 2010, pp. 1–8.

[9] D. L. Lewis and H.-H. S. Lee, "A scanisland based design enabling prebond testability in die-stacked microprocessors," in *Proc. IEEE Int. Test Conf.*, Santa Clara, CA, Oct. 2007, pp. 1–8.

[10] A. Kumar, S. M. Reddy, I. Pomeranz, and B. Becker, "Hyper-graph based partitioning to reduce DFT cost for pre-bond 3D-IC testing," in *Proc. Design Autom. Test Eur. Conf. Exhib.*, Grenoble, France, 2011, pp. 1–6.

[11] P.-Y. Chen, C.-W. Wu, and D.-M. Kwai, "On-chip TSV testing for 3D IC before bonding using sense amplification," in *Proc. Asian Test Symp.*, Los Alamitos, CA, 2009, pp. 450–455.

[12] C. Giri, S. K. Roy, B. Banerjee, and H. Rahaman, "Scan chain design targeting dual power and delay optimization for 3D integrated circuit," in *Proc. Int. Conf. Adv. Comput., Control, Telecommun. Technol.*, Kerala, India, 2009, pp. 845–849.

[13] J. L. Bentley, "Experiments on traveling salesman heuristics," in *Proc. ACM-SIAM Symp. Discrete Algorithms*, Philadelphia, PA, 1990, pp. 91–99.

[14] D. Eppstein, "Fast hierarchical clustering and other applications of dynamic closest pairs," in *Proc. ACM-SIAM Symp. Discrete Algorithms*, 1998, pp. 619–628.

[15] Y. C. Hu, Y. L. Chung, and M. C. Chi, "A multilevel multilayer partitioning algorithm for three dimensional integrated circuits," in *Proc. 11th Int. Symp. Qual. Electron. Design*, San Jose, CA, 2010, pp. 483–487.

[16] M.-K. Hsu, Y.-W. Chang, and V. Balabanov, "TSV-aware analytical placement for 3D IC designs," in *Proc. ACM 48th Design Autom. Conf.*, 2011, pp. 664–669.