Control Constrained Resource Partitioning for Complex SoCs*

Dan Zhao† Shambhu Upadhyaya† Martin Margala‡‡

†*Dept. of Computer Science and Engineering*
University at Buffalo (SUNY)
{*danzhao,shambhu*}*@cse.buffalo.edu*
‡‡*Dept. of Electrical and Computer Engineering*
University of Rochester
margala@ece.rochester.edu

Abstract

When moving into the billion-transistor era, the wired interconnects used in conventional SoC test control models are rather restricted in not only system performance, but also signal integrity and transmission with continued scaling of feature size. On the other hand, recent advances in silicon integrated circuit technology are making possible tiny low-cost transceivers to be integrated on chip. Based on the recent development in "radio-on-chip" technology, a new distributed multihop wireless test control network has been proposed. Under the multilevel tree structure, the system optimization is performed on control constrained resource partitioning and distribution. Several system design issues such as radio-frequency nodes placement, clustering and routing problems are studied, with the integrated resource distribution including not only the circuit blocks to perform testing, but also the on-chip radio-frequency nodes for intra-chip communication.

1: Introduction

A System-on-Chip (SoC) is designed by reusing pre-designed, pre-verified IP cores on a single silicon, where the system can be viewed as an interconnected network of various functional modules. With continued scaling of microelectronics, more complex systems, utilizing several hundreds of embedded components, will be placed on a single chip and today's SoC will become tomorrow's IP core. Testing such high-density high-volume core-based SoCs faces three major issues: *accessing deeply embedded cores with high-speed high-efficiency low-cost interconnect structure; partitioning test resources and scheduling IP cores to achieve maximum parallelism; and developing a high-efficiency low-cost control network to execute the test application based on a predetermined schedule*. Various test scheduling and wrapper/TAM (test access mechanism) optimization algorithms have been proposed in the literature [1, 2, 12, 13, 14, 16, 17, 18] to reduce test cost in terms of test application time. However, less attention is paid to test control cost which constitutes a major part of the total test overhead [15].

Currently, the control network connects the system level controller with local control mechanisms by wires in one of the three structures: star, bus, and multiple bus. In order to surpass the fundamental limitation of conventional hard-wired metal interconnects and advance the development of future ultralarge-scale integrated systems (ULSIs), a new radio frequency (RF)/Microwave interconnect technology has been introduced for future intra-chip communications [3, 7]. In [7], the feasibility of employing on-chip wireless interconnects for clock distribution has been investigated.

* Research supported in part by Microelectronics Design Center, University of Rochester through a grant from NYSTAR.

The tiny receivers, transmitters and on-chip zigzag antennae are implemented in 0.18μm TSMC CMOS technology with area consumption of 0.116mm^2, 0.215 mm^2 and 0.15mm^2, respectively. In particular, for a die size of 2.5cm microprocessor, the total area with one transmitter, 16 receiver and 17 antennae will consume about 1% area [6]. With the integration of tiny antennae and transceivers onto a single chip, the chip-based wireless radios can replace the wires used in conventional control network to increase accessibility, to improve bandwidth utilization, and to eliminate delay and cross-talk noise in conventional wired interconnects.

One major problem in the wireless test control architecture is to design a test procedure to ensure quality testing of all IP cores in minimum testing time and the associated test control cost. Several system optimization issues, such as RF nodes distribution (which carry out the control signals chipwide), core clustering, and routing (wireless routing as well as hard-wiring between the IP cores and their dedicated RF nodes), are brought forward for the control constrained resource partitioning and distribution. In this work, we present the optimization technique for the integration of resource distribution including not only the circuitry to perform testing, but also the radio frequency links for intra-chip communication. We assume that a hierarchical multihop test control architecture is used to achieve the most possible parallelism. In Sec. 2, we first present a brief overview of our previously proposed wireless test control architectures. Then, we study several system optimization problems such as RF node placement, clustering and routing in Sec. 3, and present a routing cost optimization scheme which minimizes the overall test control cost for SoCs in Sec. 4. Then in Sec. 5, we introduce the integrated system resource optimization. Finally, we conclude the paper and present the future work in Sec. 6.

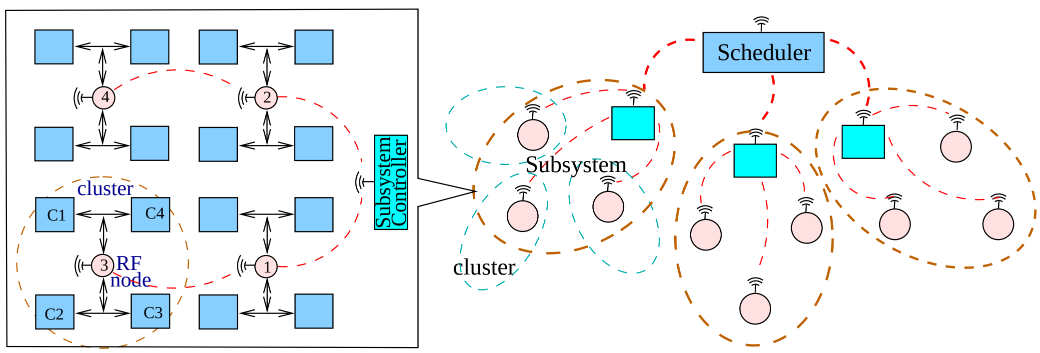
2: Overview of Wireless Test Control Architectures

We have proposed three types of wireless test control architectures in [19], i.e., miniature wireless LAN (mini-WLAN), multihop wireless test control network (MTCNet), and distributed multihop scheme (D-MTCNet). The proposed architectures consist of three basic components, the test scheduler, the resource configurators, and the radio frequency nodes supporting the communication between the scheduler and the IP cores.

The test scheduler is a central controller, which controls the resource configurators and communicates with IP cores through the RF nodes and also with the chip external. The scheduler implements the test scheduling algorithms to avoid possible conflict arisen during resource utilization and test application. The resource configurator controls and initializes the test resources. The IP cores are organized into clusters, and one or several cores with similar functionality are associated with the same resource configurator. In the case when more than one tests share common test resources, the configurators are activated (on command of the scheduler) such that no conflicts resulted in the use of resources. The RF nodes with tiny low-cost on-chip transceivers that can (two-way) communicate with the scheduler through (possibly multihop) wireless links are employed to provide wireless connections for the billion transistor SoCs, where the wired interconnects encounter fundamental limitations. While it is possible to equip each core with a dedicated RF node, it is more feasible to assign one RF node to each cluster of cores in order to reduce the cost due to the area and power overhead. IP cores are hard-wired to the RF node of its cluster, which has the RF interface.

A miniature wireless LAN was initialy proposed to work as the intra-chip test control network for SoCs, where the scheduler broadcasts control signals to RF nodes among clusters of cores. With this simple design, all RF nodes should be within the transmission range of the scheduler, and the interference between RF nodes need to be carefully concerned. In addition, the transmission power grows with the transmission range to the power of 2 to 4, and the heat dissipated by using higher power transmission may damage the surrounding circuitry. Therefore, we have proposed a low-power, high-efficiency multihop wireless test control network (MTCNet) [19], where multiple hops may be needed for one RF node to exchange data with another across the network. For example, as shown in Figure 1a, RF nodes 1 and 2 are within the direct wireless transmission range of the scheduler, and transmission of control signals between node 3 (or 4) and the scheduler is through node 1 (or 2). Thus, some nodes (for instance, node 1 or 2) operate not only as a host but also as a router, forwarding signals to other clusters in the network.

In order to improve parallel test control processing, we have further proposed an advanced hierarchical multihop scheme (D-MTCNet) [19]. In D-MTCNet, the scheduler is the system controller coordinating a set of subsystem controllers which are distributed within the transmission range of the scheduler. Each subsystem includes a number of clusters and has similar architecture as the basic MTCNet as shown in Figure 1. In this multilevel tree structure, the system controller will send the control information to the subsystem controllers which in turn control their subnetwork, such that efficient parallel communication is achievable.



(a) Basic MTCNet (b) Advanced D−MTCNet

Figure 1. The distributed multihop architecture.

3: Cost-Oriented Resource Distribution

The goal of designing a test procedure is to minimize the overall testing cost including both the test application time and the hardware cost associated with the testing process. In this section, we will propose an integrated optimization scheme to efficiently distribute RF nodes on chip and minimize the associated test control cost. Given an SoC embedded with n cores and m test resources, the test sets parameters to perform testing, and a tentative floor plan of the cores, we determine the optimal distribution of RF nodes which provide on-chip wireless communication, and minimize the overall routing cost induced by the wireless test control network.

3.1: System Modeling

We consider an SoC with a tentative floor plan, where a core c_i has the coordinates (x_i, y_i). Each RF node has a maximum assistant distance R, within which it can connect to the cores. The *RF node distribution problem* (P_{RD}) is to determine the optimal placement of RF nodes to minimize the overall routing cost on chip, such that (1) the optimal number of RF nodes is obtained, (2) the RF nodes are optimally placed to ensure all cores are within the maximum assistant distance of at least one RF node, and (3) the cores are optimally clustered. The P_{RD} problem can be formulated into *geometric disk covering* [9, 11], where a (clustered) wireless control network can be abstracted as a set of disks, each centered at a RF node with a radius of R, that covers a set of embedded IP cores (wireless clients) in the chip plane. It is known that this problem is strongly NP-complete [8].

A graph $G(V, E)$ is used to represent the SoC. V is a set of vertices, each representing an embedded core. E is a set of edges, each connecting two vertices within a distance of $2R$. Assuming an optimally placed RF node can assist at least two IP cores, i.e., a disk covers at least two vertices (it becomes trivial if one RF node can assist only one core), we can always move the RF node in a way that ensures two vertices on the border of the disk while covering the same set of vertices (see Figure 2a) [11]. Since there are two ways to place the RF node through the two vertices connected by an edge (see Figure 2b), we only have to consider maximum $2|E|$ possible disk placements. The position of each disk is specified by its center, i.e., each RF node placement $C_i(X_i, Y_i)$ ($1 \leq i \leq 2|E|$). By formulating the RF nodes distribution into disk covering, we can apply existing heuristic and approximation algorithms to solve it [4, 5, 10, 11]. In [19], we have discussed a greedy heuristic with a worst case error ratio bounded by $(ln|M| + 1)$. Here we briefly introduce an approximation approach that has a constant error ratio.

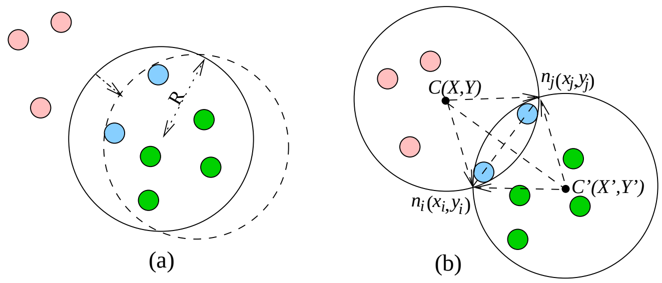


Figure 2. The illustration of disk covering.

3.2: Grid Disk Covering Scheme

Hochbaum and Maass have presented in [11] a polynomial approximation scheme that applies a shifting technique in the context of planar graphs. Separately, Franceschetti et al. [10] have proposed a grid strategy to find a covering of the points by placing disks only at the vertices of a mesh. We further present a grid disk covering scheme using a divide-and-conquer method that combines the grid strategy with the shifting technique.

We assume the shift parameter be l. Two nested application of the shift strategy is used. First, the chip plane is cut into vertical strips of width $l \times D$ (groups of l consecutive strips of width D=$2R$ are considered). By repeating the shift of all groups l-1 times over length D, there are l different ways of partitioning of the plane into strips of $l \times D$ wide, each partitioning denoted as S_i ($1 \leq i \leq l$). Then, in order to cover the points in such a strip, the shifting strategy is applied in the other dimension. Thus, the considered strip is cut into squares of side length $l \times D$. The optimal covering of points in such a square is found by applying the grid strategy. As we can see that with $\lceil l\sqrt{2} \rceil^2$ disks of diameter of D, the square of side length $l \times D$ can be covered compactly, i.e., the number of disks to cover points in the square does not exceed $\lceil l\sqrt{2} \rceil^2$. Thus, the number of possible disk positions is finite. By checking all possible arrangement of maximum $\lceil l\sqrt{2} \rceil^2$ disks, an optimal covering is found within the square. In addition, the coverage of a RF node C_i is represented by a set of IP cores covered by the disk, denoted by P_i. Let A be the grid covering algorithm that provides optimal covering within each square. For a given partition S_i, let $A(S_i)$ be the divide-and-conquer approach that applies A to each square and outputs the union of all disks used (i.e., $\cup_{i=1}^{K} P_i$=$|V|$). The minimum cardinality of the l partitioning is chosen to be the final best covering. The proposed algorithm has a polynomial time approximation complexity of $O(l^2 \lceil l\sqrt{2} \rceil^2 (2N)^{2\lceil l\sqrt{2} \rceil^2 + 1})$ with performance ratio $\leq (1 + 1/l)^2$.

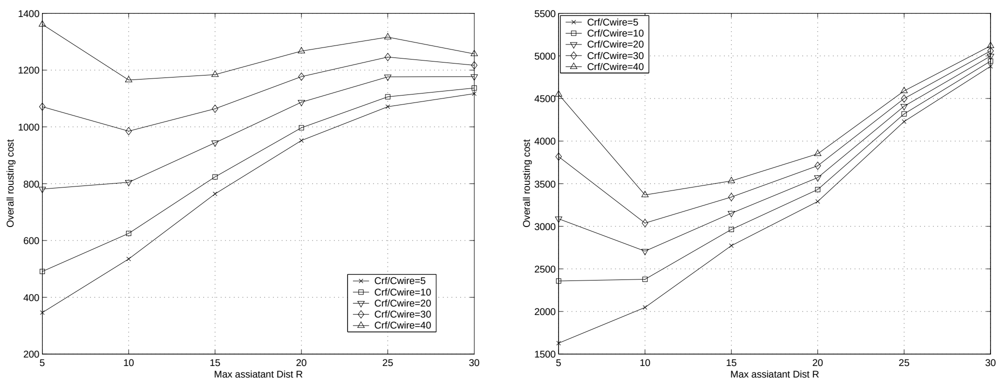
4: Routing Cost Optimization

The overall routing cost includes the wireless communication cost that is represented by the number of RF nodes and the hard-wiring cost that is represented by the total wire length between the IP cores and their associated RF nodes. The maximum RF node assistant distance determines both the number of RF nodes needed and the hard-wiring between the RF node and the cores in its cluster. Clearly, the wireless communication cost reduces at the expense of increased hard-wiring cost, and vice versa. In order to optimize the resource distribution on chip, we define a routing cost function,

$$C_{route} = C_{wire} \times L_{wire} + C_{rf} \times N_{rf}$$

where, C_{route} is the overall routing cost, C_{wire} is the unit hard-wiring cost and C_{rf} is the cost of an RF node. L_{wire} is the total wiring length between the cores and their RF nodes within the clusters. N_{rf} is the total number of RF nodes distributed chip-wide.

In simulation scenario 1, the overall routing cost with R ranging from 5 to 30 is shown in Figure 3a and 3b with 50 and 200 cores, respectively. As we can see, if the cost of an RF node is relatively low ($C_{rf}/C_{wire} < 20$), one may deploy as many RF nodes as possible to minimize the overall cost. On the other hand, when a RF node is expensive ($C_{rf}/C_{wire} > 20$), the optimal value of R (and accordingly the number of RF nodes) can be determined based on the lowest overall routing cost. For example, $R = 10$ is optimal as shown in Figure 3.



(a) C_{route} changing over R when N=50. (b) C_{route} changing over R when N=200.

Figure 3. Illustration of the overall routing cost optimization.

Further enhancement is employed by considering the core clustering and the workload balancing during the placement of RF nodes. The pseudo-code of the algorithm is illustrated in Figure 4. We first specify the clustering options due to resource constraint. For example, some cores, having similar functionality, or competing for the same resource, or physically adjacent with precedence constraint, can be grouped into the same cluster and share the same RF node. We use a virtual core c_{g_i} to represent each such group of cores labelled with the workload of the number of cores in the group. We also need to specify the hierarchical cores and use a virtual core c_{h_i} to represent a hierarchy of cores. Moreover, in order to overcome the situation where most of the cores are assigned to one or a few of RF nodes, we add the maximum workload bound, B. We estimate the length of wires between a core and its RF node using Manhattan distance function, $L(v_i, v_j) = |x_i - x_j| + |y_i - y_j|$, where v_i and v_j are two points at coordinates (x_i, y_i) and (x_j, y_j). The optimization is performed in a way that for a given maximum assistant distance of RF node R_i, we apply the grid disk covering algorithm proposed in Sec. 3.2 to find the minimum number of RF nodes needed to provide the entire wireless coverage on-chip. The selected RF nodes placement can be computed accordingly and the total hard-wire length (i.e., the sum of the length between the cores and their RF nodes) can be obtained. We compute the overall routing cost accordingly. By repeating these operations for R_i changing from l_{min} (the shortest distance between any two cores) to l_{max} (half of the chip side length), the minimum cardinality of the overall routing cost is chosen to be the final solution of RF node distribution.

```
input:  N              /* # of IP cores embedded in an SoC
        n_i(x_i, y_i)  /* floor plan of the cores
        R              /* max assistant dist of RF node
        B              /* load bound

specify clustering options by resource constraint;
initialize min(C_route);
for(l_min<=R_i<=l_max)
    determine the minimum number of RF nodes;
    determine the placement of the selected RF nodes;
    calculate C_route(R_i);
    if C_route(R_i) < min(C_route)
        min(C_route) = C_route(R_i);
output min(C_route)
output: the corresponding optimal number of RF nodes
        the optimal RF nodes placement
        the clustering of the cores
```

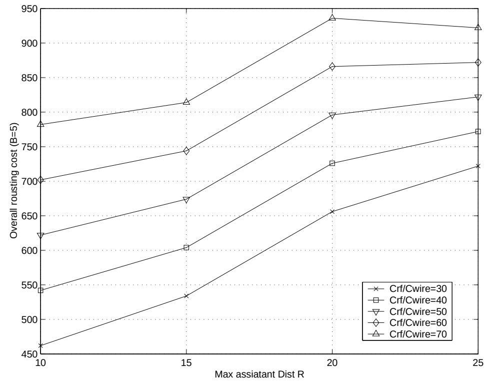
Figure 4. The routing cost optimization algorithm.

In simulation scenario 2, we study the effect of the workload balancing on the overall routing cost optimization and determine the minimum test control cost. Assuming an example SoC with 25 cores distributed on the chip with size of 50×50. We run the routing cost optimization algorithm on the SoC when R changing from 10 to 25. As shown in Figure 5, we obtain the optimally placed RF nodes and accordingly the overall routing cost. As we can see, when B is 5, the overall routing cost increases as R increases, and we obtain the lowest C_{route} when R=10. That means, we use as many RF nodes as we can to reduce the routing cost. When B increases, the C_{route} curve drops at first and then increases. When B=8, the lowest C_{route} is settled at R=20 ($C_{rf}/C_{wire} > 50$), i.e., the optimal number of RF nodes is shifted to 20.

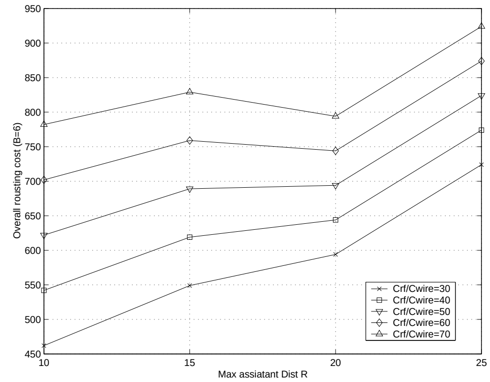
5: Integration of System Resource Distribution

The system resources in an SoC consist of two parts, the circuit blocks required to perform a test (the test resources) and RF nodes in the intra-chip wireless test control network. We have addressed the distribution of RF nodes, and the test resource distribution mainly focuses on three issues: the placement of test sources, the placement of test sinks, and the optimal routing of test access mechanisms. In this section, we present a test model (see Figure 6) for the integration of system resources in concurrent testing of core internals and externals under wireless test control.

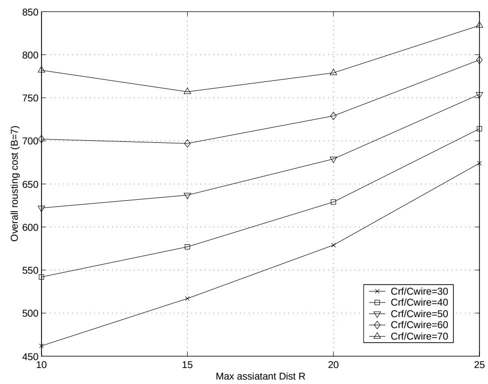
There are three types of test pattern source and sink used in this model, S_{off}, S_{self}, and S_{inter} as shown in Figure 6. S_{off} is implemented off-chip by using external ATE, and S_{self} and S_{inter} are implemented inside a chip (i.e., on-chip). S_{self} is used for testing the core itself (such as in BIST-enabled core), while S_{inter} is used for testing interconnects. We assume that each individual core is testable by either BIST or external test or a combination of them. Meanwhile, we take into consideration core testing (IEEE P1500 wrapped cores) as well as interconnect testing (which includes the testing of interconnect logics, UDLs and wiring). We assume that the IP cores in an SoC have the IEEE P1500 wrapper interface which switches between different modes, internal test mode, external test mode and normal function mode, according to the control signals received. Various test conflicts may appear during core testing and interconnect testing. Special care should be taken during the testing of a UDL (for example) and the two cores c_i and c_j which connecting the UDL. In other words, the testing of the UDL and cores c_i and c_j cannot be carried out at the same time due to wrapper usage conflict. A test conflict also occurs when the common test resources are



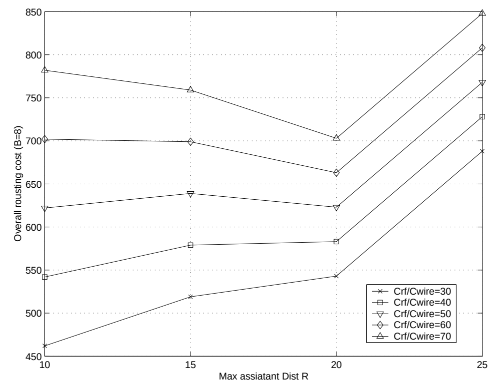
(a) C_{route} changing over R when B=5.



(b) C_{route} changing over R when B=6.



(c) C_{route} changing over R when B=7.



(d) C_{route} changing over R when B=8.

Figure 5. The overall test control cost optimization.

shared among a set of cores or the same core is tested by several test sets. Thus the test sets for the same core cannot be executed at the same time and the same resource can only to used by one test set at one time.

In this model, the test resource partitioning is directed by the wireless controlling, where each partitioning would be driven by the same set of control signals. Corresponding to each concurrent test set, F_i, a set of subsystem controllers are needed to issue a set of control signals to parallel-process the controlling and these control signals are routed along different paths. Each control signal drives different test resources required for dedicated cores in the same F_i. Our objective is to select an effective test resource partitioning that leads to the best test solution.

6: Conclusion & Future Work

One of the major objectives of SoC testing is to minimize the testing time and the associated overhead for test control. We have proposed wireless test control network using radio frequency nodes with tiny on-chip transceivers. In this paper, we have studied several system optimization issues, such as RF nodes distribution, core clustering, and routing for the control constrained resource partitioning and distribution. We have addressed the control cost minimization problem in terms of the overall routing cost. In addition, we have presented the optimization technique for the integration of resource distribution including not only the circuitry to perform testing, but also the radio frequency links for intra-chip communication. In future work, we will address the integrated

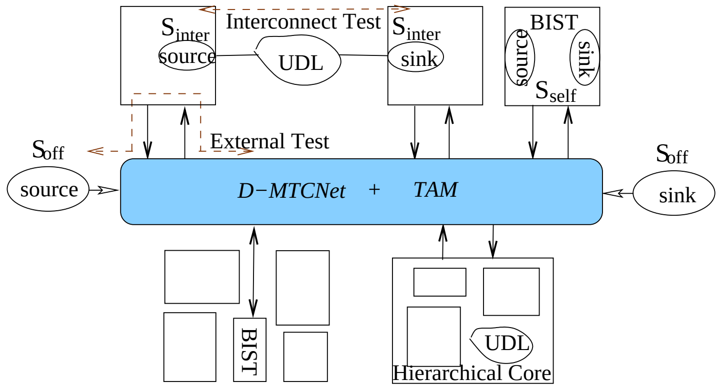


Figure 6. An Integrated Framework for Core Test and Interconnect Test.

system optimization on resource distribution and test scheduling and the impact of the wireless test control on the overall system testing solution.

References

[1] K. Chakrabarty. Design of system-on-a-chip test access architectures using integer linear programming. In *Proc. of IEEE VLSI Test Symp.*, pages 127–134, 2000.

[2] K. Chakrabarty. Test scheduling for core-based systems using mixed-integer linear programming. *IEEE Trans. on CAD of Integrated Circuits and Systems*, 19(10):1163–1174, 2000.

[3] M.F. Chang, V.P. Roychowdhury, L. Zhang, H. Shin, and Y. Qian. RF/wireless interconnect for inter- and intra-chip communications. *Proc. of The IEEE*, 89(4):456–466, April 2001.

[4] V. Chv'atal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979.

[5] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to Algorithms, Second Edition*. The MIT Press, 2001.

[6] B.A. Floyd. A CMOS wireless interconnect system for multigigahertz clock distribution, 2001. PhD Dissertation, University of Florida.

[7] B.A. Floyd, C. Hung, and K.K. O. Intra-chip wireless interconnect for clock distribution implemented with integrated antennas, receivers, and transmitters. *IEEE Journal of Solid-State Circuits*, 37(5):543–552, May 2002.

[8] R.J. Fowler, M.S. Paterson, and S.L. Tanimoto. Optimal packing and covering in the plane are NP-complete. *Information Processing Letters*, 12(3):133–137, 1981.

[9] M. Franceschetti, M. Cook, and J. Bruck. A geometric theorem for approximate disk covering algorithms. In *Paradise Technical Report, ETR035*, January 2001.

[10] M. Franceschetti, M. Cook, and J. Bruck. A geometric theorem for wireless network design optimization. In *The Lee Center for Advanced Network Workshop*, October 2002.

[11] D.S. Hochbaum and W. Maass. Approximation schemes for covering and packing problems in image processing and VLSI. *Journal of the ACM*, 32(1):130–136, 1985.

[12] Y. Huang, W.T. Cheng, C.C. Tsai, N. Mukherjee, O. Samman, Y. Zaidan, and S.M. Reddy. Resource allocation and test scheduling for concurrent test of core-based SoC design. In *The 10th Asian Test Symposium*, November 2001.

[13] V. Iyengar, K. Chakrabarty, and E.J. Marinissen. Test wrapper and test access mechanism co-optimization for system-on-a-chip. In *Proc. of ITC*, pages 1023–1032, 2001.

[14] E. Larsson and Z. Peng. Test scheduling and scan-chain division under power constraint. In *Asian Test Symposium*, November 2001.

[15] S. Misra, S. Subramanian, and P.P. Chaudhuri. A control constrained test scheduling approach for VLSI circuits. In *The 1st Asian Test Symposium*, November 1992.

[16] M. Nourani and C. Papachristou. An ILP formulation to optimize test access mechanism. In *Proc. of ITC*, 2000.

[17] D. Zhao and S. Upadhyaya. Adaptive test scheduling in SoCs by dynamic partitioning. In *IEEE Int'l Symp. on Defect and Fault Tolerance in VLSI Systems*, November 2002.

[18] D. Zhao and S. Upadhyaya. Power constrained test scheduling with dynamically varied TAM. In *IEEE VLSI Test Symposium*, April 2003.

[19] D. Zhao, S. Upadhyaya, and M. Margala. A new distributed test control architecture with multihop wireless test connectivity and communication for gigahertz system-on-chips. In *Proc. of NATW*, pages 90–93, May 2003.