Combinatorial Methods for Integrated Circuits Design

Organizers:

Thomas Lengauer (GMD, Schloss Birlinghoven/Universität Bonn)
Majid Sarrafzadeh (Northwestern University, Evanston)
Dorothea Wagner (Technische Universität Berlin)

October 18-22, 1993

The size and complexity of present day VLSI integrated circuits (IC) and systems demands the elimination of repetitive manual operations and computations in their design. This motivates the development of automatic design systems. Automation of a given (design) process requires its algorithmic analysis. The availability of fast and easily implementable algorithms is essential for the discipline, as heuristic techniques and ad hoc approaches, by themselves, cannot cope with the complexity of current (and future) IC systems. Indeed, IC designers are forced to employ algorithmic techniques and systematic approaches to design fast and reliable VLSI circuits.

The objective of this workshop was to bring together researchers whose common grounds are that they have an understanding of algorithm design, combinatorial structures, and graph theory and are interested in VLSI applications. These researchers either have been working on IC design and are applying algorithmic techniques in order to cope with the complexity of VLSI systems. Or they have been working on algorithmic graph theory and combinatorics and have found a new source of problems in VLSI domain. A focus of the workshop was on physical design, since this is the part of IC design that encompasses the most challenging applications of combinatorial algorithms.

The purpose of this workshop was to further investigate practical VLSI models, address important and relevant issues, and to apply recent developments in algorithm design and combinatorial structures to VLSI problems. One goal was to develop models that accurately practical practical issues and thus generalize some of the models currently being used. Corresponding problems need to be formulated without sacrificing their realistic nature.

During the workshop 25 lectures have been presented by the participants from different European countries, U.S.A., Japan, Taiwan and Korea. The lectures covered various topics from graph algorithms and combinatorics, e.g. Steiner trees and disjoint paths problems, cut and partitioning prob-

lems, and WQO based methods, as well as subjects from practical design as FPGA technology mapping, global optimization for practical VLSI problems and different layout systems. There were two problem sessions and an evening discussion on the *Interaction between Theory and Practice in Physical Design*.

All participants appreciated the stimulating and cordial atmosphere at Schloß Dagstuhl. The always engaged support of the Dagstuhl team was an essential contribution to the success of this seminar.

The organizers wish to thank all those who helped make the workshop an interesting and fruitful research experience.

Monday, October 18

Morning Session	Chair: Dorothea Wagner
Ernest Kuh	Timing-Driven Layout System for Cell-Based Design
Edmund Ihler	Minimum Rectilinear Steiner Trees for Intervals on Two Parallel Lines
Heike Ripphausen-Lipa	Linear-Time Algorithms for Disjoint Two-Face Paths Problems in Planar Graphs
Andrew B. Kahng	Optimal Delay Steiner Tree
Problem Session	Chair: Majid Sarrafzadeh
Afternoon Session	Chair: Martin Wong
Der-Tsai Lee	Topological Routing
Dietmar Cieslik	Steiner Minimal Trees
Dorothea Wagner	A Linear-Time Algorithm for Edge-Disjoint Paths in Planar Graphs

Tuesday, October 19

Morning Session	Chair: Yoji Kajitani
Chung-Kuan Cheng	Research on Hierarchical Partitioning
Frank Wagner	A Really Simple Min Cut Algorithm
Jun-Dong Cho	Approximation Results on Max Cut and Related Problems
Afternoon Session	Chair: Der-Tsai Lee
Michael Langston	WQO-Based Methods for VLSI-Design Theory
Pawel Winter	Euclidean Steiner Trees Problems with Obstacles
Hitoshi Suzuki	A Linear Algorithm for Finding Vertex-Disjoint Paths in Planar Graphs
Evening Session	Chair: Thomas Lengauer
Discussion:	The Interaction between Theory and Practice in Physical Design

Wednesday, October 20

Morning Session	Chair: Thomas Lengauer
Dian Zou	NST Method for VLSI Placement
Andrew B. Kahng	New Ideas in Finite-Time Global Optimization
Martin Wong	Clustering for Minimum Delay

Thursday, October 21

Morning Session	Chair: Michael Langston
Martin Middendorf	The Complexity of Manhattan Channel Routing with Single-Sided Nets
Donna Brown	The Terminal Assignment Problem for Channel and Switchbox Routing
Karsten Weihe	Towards an Object-Oriented Framework for Combinatorial VLSI-Algorithms
Problem Session	
Afternoon Session	Chair: Donna Brown
Thomas Hecker	A New Integrated Method for Global and Detailed Routing in Standard Cell Layout
Jan-Ming Ho	Assignment of Clock Drivers
Elof Frank	Interconnect and Register Allocation in Bus Architectures

Friday, October 22

Morning Session	Chair: Ernest Kuh
Michael Kaufmann	Fast $\frac{11}{8}$-Approximation for Rectilinear Steiner Trees
Majid Sarrafzadeh	On the Complexity of Technology Mapping of FPGA
Yoji Kajitani	The Totally Perfect Bipartite for Optimal Connection Block Design of FPGA

The Terminal Assignment Problem for Channel and Switchbox Routing

Donna Brown, University of Illinois at Urbana–Champaign

We consider the channel routing problem (CRP) and the switchbox routing problem (SRP) in which terminals have flexibility in placement along the shores. Specifically, we discuss a general framework for assigning terminals subject to position, order, and/or separation constraints: this is the *terminal assignment problem (TAP)*.

Previous work has shown that the TAP is $\mathcal{NP}$–complete with only position constraints [Atallah–Hambrusch, 87] or with both position and order constraints [Cai–Wong, 90]. We show that the TAP is also $\mathcal{NP}$–complete when any *one* type of constraints is allowed: only order constraints or only separation.

We present a dynamic programming solution which optimally solves the general TAP. Let O_T (O_B) represent the ordering relation on the top (bottom) shore. For problems with just position and order constraints, the running time is $\mathcal{O}(p^{c_T} q^{c_B} L)$, where p is the number of terminals on the top shore, q is the number of terminals on the bottom shore, L is the channel length, and c_T (c_B) is the number of disjoint chains in O_T (O_B). If separation constraints are included, the time is $\mathcal{O}(p^{c_T} q^{c_B} L^3)$. The solution can easily be extended to handle channels with side–exiting nets or the knock–knee model. In addition, we can solve constrained cost functions, e.g. minimizing density subject to a maximum bound on flux. The solution works for a number of cost functions, such as minimizing density, total density, maximum net span, etc.

We extend the TAP to switchboxes. For the special case in which terminals are permutable within prespecified groups of adjacent vertices, called clusters, we give an algorithm which solves the TAP whenever a solution exists. The algorithm has an $\mathcal{O}(n \log n)$ running time, where n is the number of nets, and extends to multiple layers and to convex grids. (Joint work with Jon Ramkins.)

Research on Circuit Partitioning

Chung–Kuan Cheng, University of California at San Diego

The talk covers the algorithms, the theoretical investigation and the application of partitioning problems.

1. *Algorithms:* Ratio cut, stable two–way cut, multi–way cut and cluster ratio cut.
2. *Theory:* Ancestor tree and optimal partitioning.
3. *Applications:* Multi–level partitioning for hardware simulation.

Current research on hierarchical and performance–driven partition of sequential machines is also discussed.

A Generalized Multi–Way MaxCut Partitioning

Jun–Dong Cho, Northwestern University

Graph partitioning is related to a number of VLSI layout problems that seem to have a common structure. We investigate such a structure that leads to a unified approach for a number of VLSI layout problems such as partitioning, layer assignment and placement. First we present a linear–time approximation algorithm for maxcut and two generalized problems: *maxcut k–coloring* and *maximal k–color ordering problem.*

For a graph G with e edges and n vertices, our maxcut approximation algorithm runs in $\mathcal{O}(e+n)$ sequential time yielding a node–balanced maxcut with size at least $(e + e/n)/2$, improving the time complexity of $\mathcal{O}(e \log e)$ known before. Building on the proposed maxcut technique and employing a height–balanced binary decomposition, we devise an $\mathcal{O}(e + n \log k)$ time algorithm for the maxcut k–coloring problem which always finds a k–partition of vertices such that the number of bad edges does not exceed $\frac{e}{k}(\frac{n-1}{n})^h$, where $h = \lceil \log_2 k \rceil$, thus improving both the time complexity $\mathcal{O}(enk)$ and the bound e/k known before.

The other related problem is the maximal k–color ordering problem that has been an open problem. We first prove that the problem is $\mathcal{NP}$–complete. A performance bound $e(k^2-1)/3k$ on maximal k–color ordering cost is attained in $\mathcal{O}((e+n)k^2+k^3)$ time. Those algorithms illustrate how the theoretical properties of simmax can be used to devise efficient graph partitioning algorithms. The relative simplicity of the algorithms and their computational economy are both keys to their application that involve large amounts of data. (Joint work with Salil Raje and Majid Sarafzadeh.)

Steiner Minimal Trees in Banach–Minkowski Planes

Dietmar Cieslik, Universität Greifswald

A Banach–Minkowski plane is a two–dimensional Banach space. Consider a finite set of points in this plane. Then any shortest tree which connects all these points is called a *Steiner minimal tree (SMT)*. It may contain vertices other than the points which are to be connected. Such points are called *Steiner points*. If we require that a shortest tree has at most k Steiner points, where k is a given positive integer, we get a k–SMT. If we connect pairs of given points only, we obtain a *minimal spanning tree (MST)*. Clearly,

an MST is a 0–SMT.

There are upper bounds for the degrees of the vertices of SMT and k–SMT, depending on the kind of Banach–Minkowski planes only. Particularly, the number 6 for SMT and the number 8 for k–SMT is an upper bound for any plane.

The local version of Steiner minimal trees is to find one point which minimizes the sum of distances to the given points. There are methods to construct such a point. As a consequence of all these statements, it is possible to get techniques to find a 1–SMT for a finite set of points in any Banach–Minkowski plane and, on the other hand, SMT and k–SMT in two–dimensional L_p–spaces.

Whereas an MST can be found easily, the determination of an SMT is still unknown or $\mathcal{NP}$–hard. The infimum of the ratios of the length of an SMT and the length of an MST, ranging over all finite sets, which is called the *Steiner–ratio* of the Banach–Minkowski plane, is considerably less than one. We give some lower and upper bounds for this number.

Register and Interconnect Allocation in Bus Architectures

Elof Frank, GMD and Universität Paderborn

In high–level synthesis of bus architectures register and interconnect allocation is the task of determining a minimum number of registers and buses to store all values and perform the communications between functional units.

We present several models of bus architectures and discuss the advantages and bottlenecks of each. For a linear architecture with distributed memory the optimal number of busses and registers can be computed by solving a scheduling problem of communications. An optimal and probabilistic solution is presented, using an integer programming formulation and its linear relaxation.

A New Integrated Approach to Global and Detailed Routing of Standard Cell Layouts

Thomas Hecker, Humboldt–Universität zu Berlin

An integrated package of synthesis tools for standard cell layout generation is presented.

Our major aim was to integrate the different phases and to preserve degrees of freedom (electrically equivalent pins, permutable sets of pins, cell symmetries, multiple terminals) as long and as much as possible.

The main components of the system are:

1. An algorithm for standard cell placement based on iterated global optimization and partitioning. An efficient algorithm has been adopted to standard cell design such that aspects of the global routing are considered.
2. An algorithm for global routing of cell row layouts which, one the one hand, exploits the relevant information for global routing provided by the placement step and, on the other hand, considers features of the following detailed routing step as optimization criteria. The algorithms are based on an advanced net classification and allow to push down both density and flux and to reduce vertical constraints.
3. A routing package comprising a set of routers organized similar to experts of a black–board system. The selection of methods (experts) is supported by a set of characteristical features of the task, called "visiting card".
4. A cell design allowing for a new over–the–cell routing scheme.

(Joint work with Ines Peters and Michael Weber.)

Assignment of Clock Drivers

Jan–Ming Ho, Academia Sinica, Taiwan

We consider the problem of assigning clock drivers with bounded driving capability for distribution of clock signals. The objective is to minimize the maximum service radius and total wiring length. To solve the problem, we restrict the points to tree topology, e.g., a minimum spanning tree. We then solve a bounded–fan–out p–center problem on the tree to cluster the points. An $\mathcal{O}(n \log^2 n)$ algorithm is given for solving this problem. Experimental results show that this approach is promising.

Minimum Rectilinear Steiner Trees for Intervals on Two Parallel Lines

Edmund Ihler, Universität Freiburg

We consider a generalization of the rectilinear Steiner tree problem, where our input is sets (called *classes*) of required points instead of simple required points. Our task is to find a minimum rectilinear tree, connecting at least one point from each class. We prove that the version, where all required points lie on two parallel lines, called *rectilinear class Steiner tree (channel) problem*, is $\mathcal{NP}$–hard. But we give a linear time algorithm for the case,

where the points of each required class are clustered, and the classes consist of non–overlapping intervals of points.

Optimal–Delay Rectilinear Steiner Trees

Andrew B. Kahng, University of California at Los Angeles

In practice, *Elmore delay* is a popular, high–fidelity routing objective. We study *optimal–delay* rectilinear Steiner trees which minimize any given linear combination of sink Elmore delays. We prove two main results:

1. Hanan's theorem generalizes to Elmore–optimal trees,[1] and
2. a new "peeling" characterization of optimal Steiner trees, where an optimal tree T^* can be decomposed into a sequence of subtrees $T_0 = \{\text{source} \equiv n_0\}$, T_1, $T_2, T_3, \ldots, T_k = T^*$ where each T_i is formed by adding some new sink by a shortest connection to an edge of T_{i-1}.

The proof of these results rely on concavity of Elmore delay at all sink nodes; a counterexample for Hanan's theorem for minimization of maximum sink delay suggests that concavity was essential to the result. In practice, these results enable a (finite) enumerative technique for finding Elmore–optimal routing trees; interestingly we find that the "Elmore routing tree" approach [Proc. ACM/IEEE Design Automation Conf. '93] is surprisingly close to optimal on average. Future directions include characterizing objectives for which Hanan's theorem holds, and investigating consequences of the "peeling" characterization of optimal Steiner trees. (Joint work with Kenneth D. Boese.)

Recent Ideas in Finite–Time Optimization

Andrew B. Kahng, University of California at Los Angeles

Large instances of intractable optimizations arise in every phase of VLSI system synthesis. We survey some recent studies of *practical* optimization.

First, we consider the model of how optimization actually takes place. In particular, optimization methods return the "best–so–far" solution in practice. This has significant implications for, e.g., the "simulated annealing" (SA) heuristic. All existing SA implementations use monotone cooling temperature schedules which are motivated by an *infinite–time* Markov analysis that is concerned *only* with the *last* solution seen. However, we see that temperature schedules that optimize the best–so–far solution are not cooling.

[1] This is surprising in view of the "cost–radius" tradeoff that Elmore delay smoothly captures across the range of technologies. Note that Elmore delay in one limiting case is equivalent to tree cost (i.e., the *RSMT problem*).

This challenges much of the "physical" intuition behind current annealing practice.

Second, we have developed insights into the relationship between problem structure (representation of solutions/problem, neighborhood topology, etc.) and amenability to traditional optimization approaches (greed, multi–start). We provide empirical evidence for a "big valley" structure of local minima in standard cost surfaces for TSP and graph bisection. There are heuristic arguments for such a "globally convex" model (based on consideration of shapes of, e.g., fixed–charge transportation problem). Our observations lead to the "adaptive multi–start" approach which obtains substantial improvements over previous TSP/bisection methods.

Design of Optimum Totally Perfect Connection–Blocks of FPGA

Yoji Kajitani, Tokyo Institute of Technology

A bipartite graph with partitioned vertex sets R and L ($|R| \leq |L|$) is called *totally perfect* if there exists a perfect matching from L_S to R for any $L_S \subseteq L$ such that $|L_S| \leq |R|$. This paper introduces a class of minimum totally perfect bipartite graphs. Then it is proved that there is a minimum totally perfect bipartite graph in the class such that degrees of vertices are evenly distributed in R and L, respectively. These results are directly applied to area–effective design of connection–blocks of FPGA such that least number of switches are contained and that switches are distributed evenly among pin–lines and track–lines.

Fast $\frac{11}{8}$ approximation for Rectilinear Steiner Trees

Michael Kaufmann, Universität Tübingen

We consider the Steiner tree problem in rectilinear metric. Only recently two algorithms were found which achieve better approximations than the "traditional" one with a factor of 3/2. These algorithms with an approximation ratio of 11/8 are quite slow and run in time $\mathcal{O}(n^3)$ and $\mathcal{O}(n^{5/2})$. A new implementation reduces the time to $\mathcal{O}(n^{3/2})$. As our main result we present efficient parameterized algorithms which reach a performance ratio of $11/8 + \varepsilon$ for any $\varepsilon > 0$ in time $\mathcal{O}(n \log^2 n)$, and a ratio of $\frac{11}{8} + \frac{\log \log n}{\log n}$ in time $\mathcal{O}(n \log^3 n)$.

Tiger: A Fast Timing–Driven Gate Array/Standard

Cell Layout System

Ernest S. Kuh, University of California at Berkeley

Tiger is a cell-based timing–driven layout system, which embodies the complete layout process from placement to detailed routing, developed at UCB. A new efficient feed-through assignment algorithm, *VERT*, is introduced. The timing issue is treated in each of the important stages of layout. To our knowledge, this is the first complete timing–driven layout system ever reported by a university.

Experimentals show that *Tiger* is much faster than *TimberWolf SC 6.0*, but guarantees chip performance while achieving comparable area. *VERT* is far superior to the previous feed-through assignment tool.

WQO-Based Methods for VLSI Design Theory

Michael Langston, University of Tennessee

We review the fundamentals of algorithmic tools based on *well-quasi-ordered sets*. Several key notions are highlighted, including limits on nonconstructivity, distinctions between search and decision, and the importance of the treewidth metric.

Recent applications of these tools, especially with respect to the immersion order, are described. The focus is largely on combinatorial problems motivated by FPGA architectures. Several new challenges arise in this setting, most notably the difficulty of handling multiple edges and the lack of treewidth bounds.

This talk concludes with a brief discussion of related results, ongoing progress and open problems.

Homotopic Routing and Testing of Routability for Planar Layouts

D.T. Lee, Northwestern University

We consider the problem of single–layer routing of n two–terminal nets using rectilinear wires in the presence of rectilinear circuit modules.

Given a topological planar VLSI layout sketch with $|F|$ features and $|W|$ non–crossing wires connecting n two–terminal nets we present an $\mathcal{O}(|F|\cdot|W|)$ time and space algorithm to do detailed routing of these n nets if it exists. The algorithm consists of three phases, computing a loose homotopy stored in *spokes matrices*, computing vertex–disjoint *rubber–band equivalent (RBE)* homotopy and a detailed routing. Each phase is shown to take $\mathcal{O}(|F|\cdot|W|)$ time, improving a previously known algorithm of Leiserson and Maley, which

runs in $\mathcal{O}(|F| \cdot |W| \cdot \log(|F| \cdot |W|))$ time. (Joint work with Hsiao–Feng Steven Chen.)

The Complexity of Manhattan Channel Routing with Single–Sided Nets

Martin Middendorf, Universität Karlsruhe

The well known Manhattan channel routing problem is whether we can find for a given set of nets and a given channel with k tracks a feasible routing such that no knock–knees occur. We show

Theorem. Manhattan channel routing with two–terminal nets is $\mathcal{NP}$–complete for each of the following three cases:

1. We are given only two–sided nets each with left terminal on the bottom boundary and with right terminal on the top boundary.
2. We are given only one–sided top nets and two–sided nets with left terminal on the bottom boundary and right terminal on the top boundary. In addition, the two–sided nets have density at most 1.
3. We are given only one–sided nets such that the bottom nets have density at most 1.

Corollary. Our theorem holds also if we consider the restricted Manhattan model where doglegs are not allowed.

Linear–Time Algorithms for Disjoint Two–Face Paths Problems in Planar Graphs

Heike Ripphausen-Lipa, Technische Universität Berlin

The problem of finding pairwise vertex–disjoint paths between k pairs of terminals is $\mathcal{NP}$–hard (Lynch). However, if we restrict the positions of the terminals to at most two face boundaries, then the problem can be solved efficiently. We present a linear–time algorithm for finding k pairwise vertex–disjoint paths between terminals (s_i, t_i) where always one terminal of a pair is incident to the designated inner face boundary and the second terminal is incident to the outer face boundary. The algorithm uses two special sets of paths:

- one set of k vertex–disjoint paths starting with the outer terminals and ending as right as possible on the inner face boundary and
- a second set consisting of k vertex–disjoint paths starting with the inner terminals and ending as right as right as possible on the outer face boundary.

If a solution to the *two–face paths problem* exists, then corresponding inner and outer paths have to intersect. The algorithm determines for every pair of terminals a suitable set of intersection vertices between inner and corresponding outer paths. Then an alternating sequence of paths segments of inner and corresponding outer paths between these intersection vertices is used for concatenating inner and corresponding outer terminals. (Joint work with Dorothea Wagner and Karsten Weihe.)

On the Complexity of LUT Minimization for FPGAs

Majid Sarrafzadeh, Northwestern University

We consider *field–programmable–gate–arrays* and study the problem of look–up table minimization. We prove the problem is $\mathcal{NP}$–complete (for $k \geq 5$, where k is the number of inputs to each LUT). We present a linear time algorithm for trees and propose a heuristic algorithm for general circuits. The heuristic works very well, indeed, produces the best results known on commonly used benchmarks.

Vertex–Disjoint Paths in Planar Graphs

Hitoshi Suzuki, Tohoku University

We present a linear–time algorithm which finds vertex–disjoint paths in a given graph, each of which connects two terminals of a net, where all the terminals lie on three specified face boundaries and two terminals of each net lie on the same face boundary. (Joint work with Tomoaki Kumagai and Takao Nishizeki.)

A Linear–Time Algorithm for Edge–Disjoint Paths in Planar Graphs

Dorothea Wagner, Technische Universität Berlin

We consider the problem of finding edge–disjoint paths in an even, planar graph between terminals lying on the boundary of the outer face.

The basic result characterizing solvable instances of this problem is the Theorem of Okamura and Seymour. It says that an even instance of the problem is solvable if and only if the cut condition is satisfied. Algorithms solving the problem in $\mathcal{O}(n^2)$ time have been presented by Becker and Mehlhorn, and by Matsumoto, Nishizeki, and Saito. Kaufmann and Klär improved the running time to $\mathcal{O}(n^{5/3}(\log\log n)^{1/3})$.

We present a new algorithm solving the problem in linear time. It is

completely different from the algorithms mentioned above. The algorithm is based on "right–first search" and is very easy to implement. (Joint work with Karsten Weihe.)

A Really Simple MinCut Algorithm

Frank Wagner, Freie Universität Berlin

We present an absolutely simple formulation of a MinCut algorithm for edge–weighted graphs which was recently developed by Nagamoshi and Ibarake. That algorithm is just a bit faster than the best so far by Hao and Orlin but is conceptionally and with respect to implementation much simpler as it uses no flow techniques at all.

Our description condenses the rather complicated formulation from the original paper into a 6–line algorithm very similar to Prim's minimum spanning tree algorithm. (Joint work with Mechthild Stöhr.)

Towards an Object–Oriented Framework for Combinatorial VLSI Algorithms

Karsten Weihe, Technische Universität Berlin

In this talk, we discuss the problem of integrating many algorithms for different combinatorial VLSI problems in a large–scale framework, including graphical display, interactive features, accesses to a database, and the like. We aim at a system which allows efficient implementations of all algorithms, but nonetheless provides a convenient, "natural" interface to the user. The crucial point is that these two requirements are to be satisfied simultaneously without significant additional overhead.

Our approach to overcome this problem is by incorporating techniques from *object–oriented programming (OOP)*. We present the basic structure of our framework and some ideas for the design of all underlying data structures and their interplay. It turns out that OOP tools are not only helpful for the inherent modeling problems (for which OOP has originally been developed), but are also promising for re–using code in different contexts without loss of efficiency, which is another crucial point in the design of frameworks for algorithms in general.

Euclidean Steiner Trees with Obstacles

Pawel Winter, University of Copenhagen

We introduce the notion of *Steiner visibility graphs*. Their applicability in connection with the construction of good quality suboptimal solutions to the Euclidean Steiner tree problem with obstacles is discussed. Polynomial algorithms generating Steiner visibility graphs are presented.

Circuit Clustering for Delay Minimization

Martin D.F. Wong, University of Texas at Austin

We consider the problem of clustering a combinatorial circuit into multiple chips subject to area constraint for delay minimization. Under the unit delay model, Lawler et al. in 1966 presented an optimal algorithm to solve this problem. In 1991, Murgai, Brayton and Sangiovanni proposed a general delay model and presented a heuristic algorithm for the problem. In this talk, we show that the clustering for delay minimization problem can be solved optimally in polynomial time under the general delay model. We also discuss an extension of our work to handle both area and pin constraints.

Neighbor State Transition Method for VLSI Optimization Problems

Dian Zhou, University of North Carolina at Charlotte

A novel technique, *neighbor state transition method*, is introduced for solving a class of optimization problems often found in VLSI designs. The method utilizes the powerful means developed for the optimization in continues space to solve the optimization problems confined to discrete points. For a well known $\mathcal{NP}$–problem, *gate array placement*, the method produces an asymptotically global optimal solution in polynomial time. An asymptotically tight upper bound $(1+\mathcal{O}(1/n^{\gamma}))F^*$ is established, where F^* is the global optimum, n is the size of the problem, and γ is a constant smaller than, but arbitrarily close to one.

Open Problems

Differential Net Routing

Given: A graph $G = (V, E)$, where each edge $e \in E$ is attached a distance $d(e)$, two k-pin nets $(v_1, v_2, ..v_k)$ and $(v'_1, v'_2, ..., v'_k)$ with $v_i, v'_i \in V$, and a parameter T.
Problem: Route the nets on G using identical topology (identical sequence of pins and Steiner points) with an objective to *minimize the total wire length* subject to the constraint that the *total distance that they route on different edges* is less or equal to T.
Note that the problem is interesting if k is fixed. (Chung-Kuan Cheng)

Multimedia

Given: m servers, $s_1, s_2, ...s_m$, and n users at end s_m. Let $d(i, i+1)$ be the cost of the communication between servers s_i and s_{i+1}, and let $p_1 \leq p_2 \leq \ldots \leq p_m$ be the disk storage cost per time-unit of each server. Let $t_1 \leq t_2 \leq \ldots \leq t_n$ be the starting times the n users view the movie.
Problem: Distribute the movie originating from server s_1 with an objective to minimize the total cost.
Note that the complexity of our algorithm turns out to be $O(m^2 n^3)$ where m is the number of servers, and n is the number of users. (Chung-Kuan Cheng)

Minimum Degree Spanning Tree Subject to Total Wire Length

Consider the *minimum degree spanning tree* ($MDST$) problem which is that of constructing a spanning tree for a graph $G = (V, E)$ whose maximal degree is the smallest among all spanning trees of G. This problem is a generalization of the *Hamiltonian Path* problem and is $\mathcal{NP}$-hard. Fürer and Raghavachari have given an approximation algorithm which finds a spanning tree of degree at most $\Delta^* + 1$, where Δ^* is a lower bound on the maximum degree k. However, the algorithm only concentrates on the problem of computing a minimum-degree topology of any spanning tree, without considering the impact on wire length.

Problem: Construct an $MLDST$ whose wire length is smallest among all $MDSTs$ of G.

The problem is also $\mathcal{NP}$-hard in graphs. But, for the problem in planes (satisfying triangle inequalities) is open if it can be solved polynomially.(Jun-Dong Cho)

Bounded Radius Minimum Spanning Tree of N points in the Plane

Given: A set of N points in the plane, $p_1, p_2, ..., p_N$, with p_1 as the *source*, and a parameter R.
Problem: Find a spanning tree such that the path length on the tree from the source to each point is less than R, and the total length of the spanning tree is minimum.
(Der-Tsai Lee)

Open Problem from WQO-Based Layout Tools

Motivation: A number of FPGA partitioning abstractions can be related to the immersion order on finite graphs.

Given: A family F of graphs that (1) is a lower ideal in the immersion order and (2) has unbounded tree-width.
Problem: Can F be recognized in time $O(n^c)$ where c is some small constant?

Remarks: It is known that F can be recognized (nonconstructively) in polynomial time because the immersion order is a well-quasi-order [1] supporting polynomial-time order tests [2]. Unlike the well-known minor order, however, recognition algorithms require time exponential in the size of the largest obstruction (which, for any ideal F, is a finite but unknown value) as long as no treewidth bound is possible.

References:

[1] N. Robertson and P. D. Seymour, "Graph Minors IV. Tree-Width and Well-Quasi-Ordering," *Journal of Combinatorial Theory Series B* 48 (1990), 227–254.

[2] M. R. Fellows and M. A. Langston, "On Well-Partial-Order Theory and Its Application to Combinatorial Problems of VLSI Design," *SIAM*

Journal on Discrete Mathematics 5 (1992), 117–126.

(Michael Langston)

Discussion

In order to set in motion the discussion on the interaction between theory and practice in layout design, Thomas Lengauer posed the following intentionally provocative questions.

For the theoreticians: Do you really care about applications?

No

– If not, what do you care about in your research?

  * "natural" problems
  * ...?...Who else should care about your results?

Yes

– If yes, how much do you talk to application people?

– Do you transfer your results? How?

– How much do theoretical confines hamper your progress with respect to practical applicability?

  * Analysis
  * Restriction on methods to apply

For the practitioners: Do you really care about theory?

No

– Do you feel you understand your problems?

  * Is this not important to you? Why not?
  * How can you be sure of your method?
  * Would you be happy if a theoretician did the work for you?

Yes

– How do you incorporate theory?

Maybe

– If it is easy?