Minimum Density Interconnection Trees*

C. J. Alpert, J. Cong, A. B. Kahng, G. Robins†, and M. Sarrafzadeh‡

CS Dept., University of California at Los Angeles, Los Angeles, CA 90024-1596
† CS Dept., University of Virginia, Charlottesville, VA 22903-2442
‡ EECS Dept., Northwestern University, IL 60208-3118

Abstract

We discuss a new minimum density *objective for spanning and Steiner tree constructions. This formulation is motivated by the need for balanced usage of routing resources to achieve minimum-area VLSI layouts. We present efficient heuristics for constructing low-density spanning trees, and prove that their outputs are on average within small constants of optimal with respect to both tree weight and density. The minimum density objective can be transparently combined with a number of previous interconnection objectives (e.g., minimizing radius or skew), without affecting the solution quality with respect to these previous metrics. Extensive simulation results suggest that applications to VLSI global routing are promising.*

1 Introduction

We address a new *minimum density* objective for spanning and Steiner tree constructions in the Manhattan plane. Our work is motivated by the area minimization requirement inherent in the global routing phase of VLSI layout (the global routing phase entails construction of spanning or Steiner interconnection trees over prescribed point sets, or *signal nets*; see [13] for a survey). Traditionally, the minimum-area objective has been approximately captured by minimizing the total edgelength in the tree: since wires have a fixed width and must be routed at a fixed separation from each other, the total tree edgelength provides an obvious lower bound on the routing area that must be added to the layout. However, the grid-based structure of integrated circuit routing resources provides additional information for determining the impact of a given interconnection topology on the chip area.

1.1 Problem Formulation

For the four-terminal signal net shown in Fig. 1, the interconnection tree of Fig. 1(a) forces at least three wires to cross the dashed line, meaning that the horizontal dimension of the chip must increase by enough to accommodate these three *routing grids*.[1] In contrast, the tree of Fig. 1(b) forces the horizontal chip dimension to grow by only one routing grid (however, the vertical chip dimension will grow by two grids, as indicated by the horizontal dashed line). In view of manufacturing constraints on the maximum chip dimension, the most effective layouts are generally those which are roughly square, and this suggests balancing the horizontal and vertical routing requirements induced by the interconnection tree. As a result, we formulate the *Minimum Density Interconnection Tree* problem as follows.

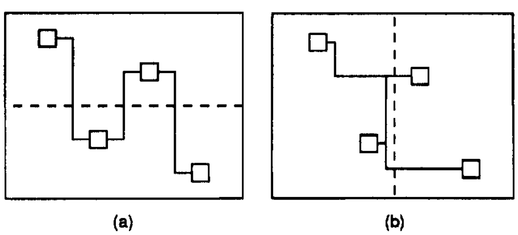

Figure 1: A four-point example for which tree (a) has density 3, but tree (b) has density 2.

A *signal net* N is a set of n *terminals*, $p_1, p_2, \ldots, p_n \in N$ in the Manhattan plane, with one terminal identified as a *source* and the rest as *sinks*. An *interconnection tree* of a net N, denoted $T(N)$, is a tree which spans N. The *cost* of a routing tree T is the sum of the costs of its edges, where the cost of an edge is the Manhattan distance between its endpoints. Without loss of generality, we assume that the terminal coordinates are scaled so that the entire signal net lies within the unit square.

Def: The *density* of an interconnection tree is the maximum number of tree edges properly intersected[2] by any horizontal or vertical line in the plane.

*Partial support for this work was provided by a Department of Defense Graduate Fellowship; by NSF MIP-9110511; by NSF MIP-9110696, NSF Young Investigator Award MIP-9257982, ARO DAAK-70-92-K-0001, and ARO DAAL-03-92-G-0050; and by an IBM Graduate Fellowship.

[1] We adopt "routing grid" as a generic term independent of layout methodology. The term encompasses, e.g., vertical feedthroughs or horizontal routing tracks in a channel [13].

[2] A line *properly* intersects an edge if and only if it intersects the edge at a single point.

0-7803-1254-6/93$03.00 © 1993 IEEE

Def: For a given net N, the *minimum density* of N is the minimum density achievable by an interconnection tree $T(N)$, and a *minimum density interconnection tree* is any $T(N)$ that achieves this density.

We will address the following:

Minimum Density Interconnection Tree (MDIT) Problem: Given a net N, find a minimum density interconnection tree $T(N)$ that has minimum cost.

1.2 Related Formulations

A number of alternative interconnection objectives trees have been examined in the VLSI CAD literature, motivated by issues of system *performance*: (i) minimizing the total tree cost (this reflects the RC delay of the wiring in addition to chip area), (ii) minimizing the maximum source-sink tree pathlength, i.e., tree *radius* (this reflects the maximum signal delay, particularly for newer interconnect technologies such as those in multi-chip module packages [8]), and (iii) minimizing the maximum *difference*, or *skew*, between source-sink pathlengths (this reflects the clock skew minimization problem.

Each of these objectives has engendered an extensive literature: the first corresponds to the minimum rectilinear Steiner tree problem [9] [12], the second has been treated in the "bounded-radius, bounded-cost" interconnection tree algorithms of [2] [4] [5] [6], and the third has been studied in, e.g., [3] [10] [11] [14]. We make note of these existing formulations because our proposed algorithms for minimum-density interconnection trees afford unique multiple optimizations wherein more than one competing objective may be addressed *simultaneously*, as discussed below.

2 Heuristics for MDIT

We assume that there are exactly $n = k^2$ terminals, and that all x and y coordinates of the terminals are distinct.

2.1 The COMB Construction

Our first algorithm partitions the terminals of net N into $\frac{\sqrt{n}}{\sqrt{2}}$ vertical strips, each containing $\sqrt{2n}$ terminals (Fig. 2a). We connect all the terminals in each strip in order of decreasing y coordinate (Fig. 2b), and then form a routing tree by joining the bottom terminals of all strips from left to right (Fig. 2c).

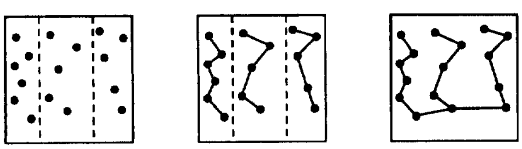

Figure 2: Sample execution of COMB.

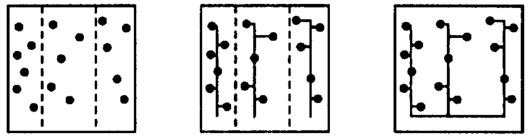

Figure 3: Sample execution of COMB_ST.

If the introduction of Steiner points is allowed, we reduce the worst-case density as well as the worst-case cost of our construction via the following method: (i) partition the net N into $\frac{\sqrt{n}}{\sqrt{2}}$ vertical strips, each containing $\sqrt{2n}$ terminals (Fig. 3a); (ii) connect all the terminals in each strip to a central *spine*[3] within the strip (Fig. 3b); then (iii) join all the spines using segments of a single horizontal line (Fig. 3c). We call this variant COMB_ST.

2.2 A Chain Peeling Method

A different, "chain-peeling" approach to density minimization iteratively computes and superposes chains or antichains. A *chain* is a sequence of terminals with coordinates that are monotone increasing in both x and y; an *antichain* has coordinates monotone increasing in x and monotone decreasing in y. A consequence of Dilworth's theorem [7] is that a point set of size n must contain either a chain or an antichain of size at least $\sqrt{n}$.

Our chain-peeling method, which we call PEEL, efficiently detects a maximal chain or antichain and then removes it from the net; the process is iterated over the remaining terminals until the net has been covered. The chains and antichains are then connected into a routing tree without increasing density. The PEEL method is attractive because it escapes such pathological examples as that of Fig. 4, where COMB or COMB_ST will yield density an unbounded factor greater than that of PEEL.

3 Performance Bounds

We can show that both the density and the total cost of our constructions are on average only small constant factors away from optimal. Proofs are omitted for brevity but may be found in [1].

Density: A lower bound of $\Omega(\sqrt{n})$ can be established for the worst-case minimum density of a spanning tree $T(N)$:

Theorem 3.1 *A net of n terminals at the grid points of a $(\sqrt{n}-1) \times (\sqrt{n}-1)$ grid cannot be spanned by an interconnection tree having density* $< \lceil \frac{\sqrt{n}+1}{2} \rceil$. ∎

[3] A *spine* is the vertical line which passes through the median terminal (ordered by x coordinate) of the strip.

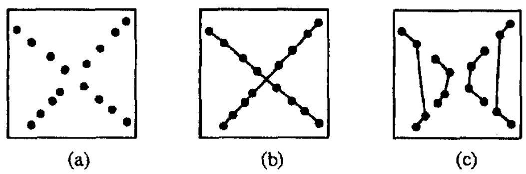

Figure 4: Class of instances (a) for which PEEL performance (b) is an unbounded factor better than that of COMB or COMB_ST (c). The connecting edges between the strips are not shown in (c). For points in an "X" configuration, PEEL always yields constant density = 2, while the COMB or COMB_ST density will grow as the square root of n.

Theorem 3.2 *For n terminals chosen from a uniform distribution in the unit square, the MDIT has expected density $\Theta(\sqrt{n})$.* □

Theorem 3.3 *Algorithms COMB, COMB_ST and PEEL construct trees $T(N)$ with densities at most $\sqrt{2n}$, $\frac{\sqrt{n}}{\sqrt{2}} + 1$ and $2\sqrt{n}$ respectively.* □

Cost: Probabilistic arguments can be used to show that on average, all of our heuristics will produce interconnection trees with low cost.

Theorem 3.4 *For n terminals chosen from a uniform distribution in the unit square, the expected cost of the minimum spanning tree is $\Theta(\sqrt{n})$.* □

Theorem 3.5 *Algorithms COMB, COMB_ST and PEEL construct trees $T(N)$ with costs at most $2\sqrt{2n}$, $\sqrt{2n} + 1$ and $4\sqrt{n}$, respectively.* □

Corollary 3.6 *For n terminals chosen from a uniform distribution in the unit square, algorithms COMB, COMB_ST and PEEL all construct trees which on average have both density and cost bounded by constants times optimal.* □

Complexity:

Theorem 3.7 *Algorithms COMB, COMB_ST and PEEL have time complexities $O(n \log n)$, $O(n \log n)$, and $O(n^{\frac{3}{2}} \log \log n)$, respectively.* □

4 Triple Optimizations

For practical VLSI routing applications, it is often desirable to minimize more than one objective function at once. However, it is usually difficult to treat even two competing measures effectively. We now show that the minimum-density objective is "compatible" with existing performance-driven routing objectives, enabling simultaneous consideration of up to three separate routing tree measures.

The work of [11] gives an iterative matching-based clock tree construction that minimizes skew while keeping total wirelength within a constant factor of optimal on average, and bounded by $O(\sqrt{n})$ always. To construct clock trees with low density, we use a variant of COMB to obtain geometric matchings with low density. Using $\sqrt{n}$ strips and joining them in a serpentine fashion yields our so-called COMB_SERP tour which has cost and density both bounded by $O(\sqrt{n})$ in the worst case.[4] The odd-numbered edges of this tour will constitute a geometric matching having both cost and density bounded by $O(\sqrt{n})$. We employ this construction within the method of [11] to yield clock routing trees that simultaneously address three measures: pathlength skew, total wirelength and density, with the last two quantities both bounded on average by constants times optimal.

Another example of a triple optimization is obtained if we combine the bounded density formulation with the BRBC cost-radius tradeoff of [6]. BRBC starts with a low-cost tour of the net terminals (e.g., a depth-first tour of a minimum spanning tree), and then augments this tour by adding shortest paths to the source from certain regularly spaced locations along this tour. The algorithm returns the shortest-paths tree over the resulting augmented graph. By using COMB_SERP as its initial tour, BRBC will construct a routing tree with radius bounded by $(1+\epsilon) \cdot R$, cost bounded by $(1 + \frac{2}{\epsilon}) \cdot 2\sqrt{2n}$, and density bounded by $(1 + \frac{4}{\epsilon R}) \cdot 2\sqrt{n}$, where $R \leq 2$ is the maximum distance from the source to the farthest sink and ϵ is a user defined parameter. These expressions imply average-case performance within constant factors of optimal for all three objectives, and the radius bound also holds in the worst case.

5 Experimental Results

We have implemented COMB, COMB_ST, and PEEL using ANSI C in the Sun environment. For each pointset cardinality, each algorithm was executed on 100 pointsets chosen randomly from a uniform distribution in the unit square. We computed the minimum, average, and maximum densities and costs of the resulting interconnection trees (see Tables 1 and 2). The average density of the tree produced by COMB is on par with the density of the minimum spanning tree, but the density of the minimum spanning tree has considerably higher variance. Thus, the COMB or COMB_SERP constructions may be desirable for their predictable performance. From the tables, we see that the average density of the trees produced by the COMB_ST algorithm is considerably better than the average density of the corresponding minimum spanning trees: for example, for $|N| = 10$, COMB_ST yields trees with average density 3.00, while the average minimum spanning tree density is 3.82. This 21% decrease in average density is achieved with a

[4] For example, the first/second strips of Fig.2c would be connected using their topmost points, and the second/third using their lowest points. The first and last points in the serpentine ordering are connected to yield the tour.

corresponding 21% increase in the tree cost over MST cost. Note that in the extended version of this work [1], we present a "computational lower bound" for a given problem instance. The method divides the unit square into an i by j (not necessarily uniform) rectangular grid such that the greatest number P of the resulting $i \cdot j$ rectangles contain terminals. In order for the tree to be connected, a tree edge must cross the boundary of each rectangle which contains a terminal. From simple counting arguments, we deduce a lower bound of $\lceil \frac{P-1}{i+j-2} \rceil$ for the density of any tree for the given problem instance. Using this lower bound, we find that COMB_ST constructs a tree with optimal density in 164 of the 200 instances for $n = 3$ and $n = 5$. Moreover, the COMB_ST outputs averaged within a factor of two of optimal for $n \leq 100$.

net	MST		PEEL		COMB_ST	
size	min/max	ave	min/max	ave	min/max	ave
3	1/2	1.69	1/2	1.69	1/1	1.00
5	2/4	2.57	2/2	2.00	2/2	2.00
7	2/5	2.97	2/3	2.66	3/3	3.00
10	2/6	3.82	2/4	3.08	3/3	3.00
15	3/6	4.35	3/5	3.93	3/3	3.00
20	4/8	4.98	4/6	4.76	4/4	4.00
30	4/8	5.99	5/7	5.88	5/5	5.00
50	5/10	7.11	7/9	7.85	6/6	6.00
100	7/12	9.48	10/13	11.48	8/8	8.00
300	12/17	14.59	19/22	20.69	13/13	13.00

Table 1: Tree density statistics for minimum spanning tree and the two heuristic constructions. Averages are taken over 100 instances for each net size.

net	MST	PEEL	COMB_ST
size	ave	ave	ave
3	1103.66	736.72	1164.96
5	1658.39	1495.57	2260.09
7	2039.34	1852.91	3009.31
10	2662.36	2776.06	3224.01
15	3224.41	3721.27	4216.83
20	3789.89	4720.69	4823.63
30	4651.00	6318.27	6570.46
50	5945.4 7	9298.73	8029.99
100	8384.32	14717.75	11083.93
300	14318.99	28960.44	18681.10

Table 2: Tree cost statistics. We omit the cost of connecting the chains and antichains in PEEL, so that the PEEL cost may be lower than MST cost.

6 Conclusions and Future Work

We have proposed a new spanning and Steiner tree formulation based on a minimum density criterion. We have also presented several efficient heuristics for constructing low-density trees. The average performance of all our algorithms is within constant factors of optimal in terms of both tree cost and density. Our techniques can also be used to unify the new density criterion with previous "performance-driven" interconnection objectives in order to achieve simultaneous optimization of up to three competing interconnection tree measures. Extensive simulations indicate that our approaches are effective in practice, and hold promise for applications to balanced-resource routing in VLSI layout.

It is still open whether there exists a polynomial-time algorithm that constructs a routing tree with both cost and density bounded by constants times optimal in the *worst* case, and whether the MDIT problem is NP-complete. Recall that PEEL holds promise in that there exist examples where it outperforms COMB and COMB_ST by a factor of $\Theta(\sqrt{n})$ (Fig. 4); we conjecture that PEEL can be shown to yield worst-case density that is within a constant factor of optimal. In fact, we offer two closely related conjectures: (i) that the minimum density of a spanning tree over net N is at least the minimum of the number of chains or the number of antichains needed to cover N; and (ii) that the PEEL algorithm will use at most two times the minimum possible number of chains/antichains that cover N.

References

[1] C. J. Alpert, J. Cong, A. B. Kahng, G. Robins, and M. Sarrafzadeh, *Minimum Density Interconnection Trees*, Tech. Rep. CS-92-35, CSD, University of Virginia, October 1992.

[2] K. D. Boese, J. Cong, A. B. Kahng, K. S. Leung, and D. Zhou, *On High-Speed VLSI Interconnects: Analysis and Design*, Proc. Asia-Pacific Conf. on Circuits and Systems, (1992), pp. 35–40.

[3] K. D. Boese and A. B. Kahng, *Zero-Skew Clock Routing Trees with Minimum Wirelength*, in Proc. IEEE Intl. ASIC Conf., Rochester, NY, September 1992, pp. 17–21.

[4] J. Cohoon and J. Randall, *Critical Net Routing*, in Proc. IEEE Intl. Conf. on Computer Design, Cambridge, MA, October 1991, pp. 174–177.

[5] J. Cong, A. B. Kahng, G. Robins, M. Sarrafzadeh, and C. K. Wong, *Performance-Driven Global Routing for Cell Based IC's*, in Proc. IEEE Intl. Conf. on Computer Design, Cambridge, MA, October 1991, pp. 170–173.

[6] ———, *Provably Good Performance-Driven Global Routing*, IEEE Trans. on Computer-Aided Design, 11 (1992), pp. 739–752.

[7] R. P. Dilworth, *A Decomposition Theorem for Partially Ordered Sets*, Ann. Math, 51 (1950), pp. 161–166.

[8] C. Hilbert and C. Rathmell, *Multichip Modeuls: System Advantages, Major Constructions, and Materials Technologies*, IEEE Press, 1991.

[9] F. K. Hwang, D. S. Richards, and P. Winter, *The Steiner Tree Problem*, North-Holland, 1992.

[10] M. A. B. Jackson, A. Srinivasan, and E. S. Kuh, *Clock Routing for High-Performance IC's*, in Proc. ACM/IEEE Design Automation Conf., 1990, pp. 573–579.

[11] A. B. Kahng, J. Cong, and G. Robins, *High-Performance Clock Routing Based on Recursive Geometric Matching*, in Proc. ACM/IEEE Design Automation Conf., June 1991, pp. 322–327.

[12] A. B. Kahng and G. Robins, *A New Class of Iterative Steiner Tree Heuristics With Good Performance*, IEEE Trans. on Computer-Aided Design, 11 (1992), pp. 893–902.

[13] B. T. Preas and M. J. Lorenzetti, *Physical Design Automation of VLSI Systems*, Benjamin/Cummings, Menlo Park, CA, 1988.

[14] R. S. Tsay, *Exact Zero Skew*, in Proc. IEEE Intl. Conf. on Computer-Aided Design, Santa Clara, CA, November 1991, pp. 336–339.