Congestion Reduction in Traditional and New Routing Architectures

Ameya R. Agnihotri Patrick H. Madden
State University of New York at Binghamton
Computer Science Department pmadden@cs.binghamton.edu

ABSTRACT

In dense integrated circuit designs, management of routing congestion is essential; an over congested design may be unroutable. Many factors influence congestion: placement, routing, and routing architecture all contribute. Previous work has shown that different placement tools can have substantially different demands for each routing layer; our objective is to develop methods that allow "tuning" of interconnect topologies to match routing resources.

We focus on congestion minimization for both Manhattan and non-Manhattan routing architectures, and have two main contributions. First, we combine prior heuristics for non-Manhattan Steiner trees and Preferred Direction Steiner trees into a hybrid approach that can handle arbitrary routing directions, via minimization, and layer assignment of edges simultaneously. Second, we present an effective method to adjust Steiner tree topologies to match routing demand to resource, resulting in lower congestion and better routability.

Categories and Subject Descriptors

J.6 [**Computer-Aided Engineering**]: CAD

General Terms

Algorithms

Keywords

Global routing, non-Manhattan design, Steiner trees, congestion

1. INTRODUCTION

Congestion minimization for dense integrated circuits has been widely studied. Informally, *congestion* is the ratio of routing demand (interconnect wire) to available routing resource (open space); an over congested design may be unroutable, or can require long routing detours that impact circuit speed.

In this paper, we study routing congestion for both traditional Manhattan routing architectures, and also for non-Manhattan architectures that have recently attracted interest. We focus on methods to optimize tree topologies and layer assignments to obtain low total wire length, low numbers of vias, and routing demand that matches available resources.

The remainder of this paper is organized as follows. We first briefly discuss routing architectures, Steiner tree heuristics, and global routing. We also describe congestion metrics and the interactions between routing architectures and placement. Next, we present our main contributions: a combined non-Manhattan Preferred Direction Steiner tree heuristic and an approach to adjust Steiner tree topologies such that routing demand matches available resources. We conclude with a summary and a discussion of current and future work.

GLSVLSI'03, April 28–29, 2003, Washington, DC, USA.

2. PRIOR WORK

There is an abundance of work on circuit routing and interconnect optimization. We briefly summarize this work.

Routing Architectures

An obvious issue in congestion minimization is the choice of routing architecture. In traditional design, individual layers have "preferred direction;" all (or almost all) wires on a given layer are oriented either horizontally or vertically. The directions on each layer usually alternate, and the minimum width wire on the lower layers is usually thinner than on upper layers.

While Manhattan routing architectures have dominated circuit design, there is growing interest in non-Manhattan architectures. First proposed as a method to address modern interconnect delay problems in [10], non-Manhattan routing is now championed by the *X Initiative*[1], an industry consortium that is advocating routing at 45 degree increments (the "X Architecture"). We use the terminology of [10], and refer to this as "octilinear" routing. In [10], routing at 60 degree increments was also discussed; there is also growing interest in this "hexagonal" routing approach[6].

With all routing metrics, there is the requirement of vias between layers. These vias have non-zero cost, and connections from lower layers to upper ones create obstacles on the intermediate layers. Each routing metric offers different routing resources; tuning routing demand to match resource is one focus of this paper. The three routing architectures we focus on are illustrated in Figure 1.

Steiner Algorithms and Layer Assignment

The problem of interconnecting a set of points on an integrated circuit is a variation of the classic Steiner problem. Well known methods for this include the 1-Steiner heuristic[9] and the edge-removal tree method[4]. Both works obtain Steiner trees that are close to optimal in practice. While the underlying problem is NP-Complete, an approach that can handle relatively large planar problems has been developed[11].

Most Steiner tree algorithms use a planar formulation, and are integrated into routing applications by adding a "layer assignment" step. In general, "global" wires are placed on upper layers, while

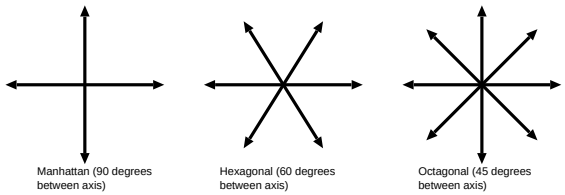

Figure 1: Manhattan, hexagonal, and octagonal routing metrics. While the bulk of VLSI routing is performed using a Manhattan metric, other routing architectures are possible.

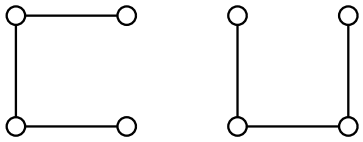

Figure 2: Routing demand for a set of points can vary widely depending on the tree topology. While tree lengths are equal, the routing demand on a given layer can vary by a large amount.

"local" wires reside on lower layers. There is good motivation for this approach: by placing longer wires on upper layers, the available resource of those layers can be used without requiring large numbers of vias These issues are discussed in greater detail by [12]. The edge-removal heuristic[4] has been adapted to simultaneously consider layer assignment and via cost[15].

While the bulk of work related to circuit design has focused on rectilinear formulations, some results have been obtained for non-Manhattan routing architectures[10, 8, 11].

Congestion Minimization in Global Routing

Congestion minimization is one of the primary objectives of global routing. Each connection may have multiple possible paths, and by selecting an appropriate set, the routing demand in any given area can be reduced.

Many global routing tools share a common framework; a shortest path algorithm (such as Dijkstra's[7]) can be used to find a path for a single connection. If routing demand exceeds resources, "rip-up and reroute" is applied. Recent work on multicommodity flow problems has resulted in an effective approximation algorithm for this problem[3].

In some formulations, the routing graph is formulated as a planar set of "tiles," with layer assignment being performed as a post-processing step. Directly considering layers during global routing increases the size of the routing graph considerably, but more accurately measures routing demand and resources.

3. NON-MANHATTAN PREFERRED DIRECTION STEINER TREES

Our first contribution is the combination of two existing heuristics for Steiner tree construction into a method which handles non-Manhattan routing architectures, layer direction restrictions, and varying via costs directly. To optimize interconnect trees for the available routing resources, we are interested in **minimum cost** Steiner trees, and not necessarily **minimum length.** In our routing model, and throughout this paper, we assume that each interconnect layer has a routing cost, and that the cost of an edge is the product of the length and the routing cost of it's layer.

The foundation of the approach is the edge-removal tree heuristic of Borah, Owen, and Irwin[4]. As the extension is fairly straightforward, we describe it only briefly.

The underlying Steiner tree heuristic begins with a minimum spanning tree, and then applies a series of transformations. By examining pairings of a single edge to a single vertex, *candidate merge* operations can be identified; if a new edge is introduced between the vertex and existing edge, a cycle is generated. By removing the longest edge on the generated cycle, total tree cost can sometimes be reduced.

This heuristic was originally developed for Manhattan routing metrics; in [10], observations originally made in [14] allow for the identification of possible Steiner point locations for non-Manhattan metrics were determined. This results in a simple extension to novel routing architectures. Recently, two other efficient methods for non-Manhattan Steiner tree construction have been developed[11, 8]. A three-dimensional Manhattan routing model with non-zero via costs[15] was another extension of the original algorithm.

To handle multiple routing layers and non-Manhattan metrics, we have combined heuristics for preferred direction Steiner trees[15] and non-Manhattan Steiner trees[10]. This requires a slight modification to the "merge" step of the edge-removal heuristic[4]. Each candidate Steiner point will join three existing points using two straight segments, and at most one edge with a single bend; layer assignments for these segments can be found using simple enumeration.

The algorithmic complexity of the heuristic is $O(n^2)$; consideration of layer assignments only contributes a constant factor to the run time (assuming a fixed and relatively small number of routing layers).

4. LAYER BALANCING FOR CONGESTION REDUCTION

Motivation for the work presented in this section comes from the following: when we consider the routing demand of placements produced by different tools, there can be surprisingly large variation. In [2], the routing demand for three placement tools Feng Shui 1.5[16], Capo[5], and Dragon[13], were compared. While the horizontal and vertical demand of Dragon placements were roughly equal across a series of benchmarks, the other two placement approaches had heavily biased demands. The ratio of horizontal to vertical demand was as much as 70:30; for successful routing, it is clear that the global router must construct Steiner trees that are as economical as possible with horizontal tree segments.

To address this issue, we present a "layer balancing" approach which provides a simple method to smoothly adjust interconnect topologies such that they better match the available routing resources. Obviously, we wish to avoid increasing total wire length or via counts substantially.

Figure 2 illustrates the type of optimization opportunity we wish to exploit. Given four equidistant points as shown, there can be two different minimum length Steiner trees; one which uses two vertical segments and a single horizontal segment, and a second which uses two horizontal segments and one vertical segments. If we assume that one interconnect layer is used for horizontal wiring, and a second for vertical, the congestion impact and routing demand of these trees can be quite different. Traditional Steiner heuristics focus on wire length minimization alone, and provide no means for selecting the appropriate topology; via costs are frequently ignored. For circuit routing, however, the "best" topology cannot be selected independently from the other interconnect nets (each of which contributes some routing demand), or from the available routing resources.

Placer	Iteration	MST Cost	Steiner Cost	MST Length	Steiner Length	MST Vias	Steiner Vias	Percent Demand Metal1	Metal2	Metal3	Metal4
IBM01											
Feng Shui 1.5	0	84289140	75073323	5903162	6061562	131036	115370	54.81	45.19	0	0
	1	84289187	75268821	5903162	6037938	131040	115740	54.93	44.95	0.12	0
	4	84732348	75696620	5903573	6057871	131798	116438	47.5	39.95	7.44	5.11
	9	85329582	76153536	5909071	6174088	132794	117011	41.03	36.64	14.09	8.25
Dragon	0	84054889	74468156	5435196	5654348	131036	114695	49.6	50.4	0	0
	1	84054889	74489791	5435196	5624468	131040	114785	49.61	50.24	0.15	0
	4	84375366	74778978	5435615	5655829	131590	115234	44.26	46.27	5.28	4.19
	9	84891510	75159463	5438669	5765054	132454	115692	39.76	41.78	9.85	8.62
IBM04											
Feng Shui 1.5	0	235792269	211111525	19476376	19780858	253712	224412	58.97	41.03	0	0
	1	235824685	211533501	19476376	19702252	253754	225003	57.79	40.92	1.28	0
	4	238109963	213845106	19479767	19756748	256430	227650	41.88	37.81	17.32	2.99
	9	239716491	215075654	19485775	20013354	258310	228795	37.93	33.82	21.16	7.09
Dragon	0	235486620	207627206	18297174	18769306	253712	220620	49.06	50.94	0	0
	1	235541417	207920191	18297174	18705296	253778	221039	47.51	50.91	1.57	0
	4	237164426	209603486	18300777	18770870	255670	222929	39.21	45.35	9.94	5.5
	9	239596061	211733302	18318027	19083897	258490	225051	32.98	39.5	16.32	11.2

Table 1: Routing demand after a number of iterations of layer balancing, using a Manhattan routing metric. Initially, routing demand is for lower routing layers; as routing cost on the lower layers is increased, some tree edges migrate towards the upper layers. MST and Steiner cost combine the cost of a routing layer and the via costs; the length columns consider actual length alone. Shifting routes to upper layers results in an increase in the total tree cost, tree lengths, and via counts.

To address congestion, and optimize routing demand to the available resources, we have developed a relatively simple approach that consists of only a few steps. We use the (non-Manhattan) Preferred Direction Steiner tree heuristic, and then iteratively adjust layer cost to either encourage or discourage routing on a given layer. The steps are as follows.

- Routing costs for all layers are set to initial default values.
- Preferred Direction Steiner trees are constructed all signal nets.
- Resource usage is examined; if a layer is overutilized, the cost of that layer is increased slightly, and we repeat the second step.

Table 1 shows the effect of adjusting layer cost repeatedly to obtain different layer assignments, using a Manhattan routing metric. We have "target" routing demand levels for interconnect layers 1 through 4 that are 30%, 30%, 20%, and 20%; our optimization attempts to place only a percentage of all wiring on a given level. The first iteration of Steiner tree construction places all connections on the lower layers; as routing cost changes, connections shift to upper layers.

4.1 Layer Cost Adjustment

As was mentioned, we are interested in minumum cost Steiner trees; by adjusting layer cost, we can alter the topologies and layer assignments of trees, resulting in changes to routing demand. Our method repeatedly adjusts the costs of a routing layer based on the demand observed in a previous iteration. If layer costs change dramatically with each iteration, the routing demand exhibits a "ringing" effect; less aggressive cost adjustment results in smooth (but slow) convergence.

While we focus on achieving a particular percentage of routing on a level, we can easily adapt the method to target fixed total lengths (which would be appropriate if accurate estimates of routing resources are available).

4.2 Impact of Via Cost

Via cost influences tree topologies in surprising ways. In Table 1, the Steiner tree length can be greater than the spanning tree length; our Steiner heuristic chooses constructions that eliminate vias substantially, some times at the expense of additional tree length.

The "cost" of a via impacts the layer balancing in the following manner. If via cost is high, the Steiner heuristic will only place very long edges on the upper layers (and only if the upper layers have lower routing cost). The cost of vias also considerably impacts the speed of convergence; if via cost is high, there can be a significant barrier to moving a segment from layer i to layer j.

The first iteration of our method shows this effect; initially, routing demand is for the lower layers, and only after reducing the cost of upper layers does the Steiner heuristic migrate edges on lower layers to upper ones.

4.3 Non-Manhattan Routing Architectures

Due to space constraints, we report only results on octilinear ("X") routing architectures; our methods have also been appiled to hexagonal routing metrics.

In our experiments with Manhattan routing architectures, we have considered two horizontal and two vertical routing layers; because there are two layer choices for any segment, slight changes in routing cost can result in a connection shifting from one layer to another. Thus, for many trees, there may be multiple topologies that have similar total length, but different layer assignments.

For non-Manhattan architectures, however, balancing of layers is more difficult. Consider the case of a connection which uses a horizontal layer; shifting to another layer would require using (perhaps) segments on both diagonal layers; this incurs a substantial increase in route length, and also substantially more vias. We find that it is more difficult to "tune" routing demand in a non-Manhattan model to the available resources, and thus it is important that a placement tool consider these resources directly.

Table 2 shows the convergence of layer demand using an octilinear routing model. As routing cost changes on each layer, connections move from the Manhattan routing layers to those that are at 45 degrees; without increasing the total tree cost substantially (and reducing the via cost as well), obtaining uniform usage of routing layers is difficult.

4.4 Window-Based Optimization

The experiments we report here focus on smaller benchmarks. We have applied these techniques to all 18 "IBM" benchmarks, as

Placer	Iteration	MST Cost	Steiner Cost	MST Length	Steiner Length	MST Vias	Steiner Vias	Percent Demand Metal1	Metal2	Metal3	Metal4
IBM01											
Feng Shui 1.5	0	84328887	75105664	5872098	6033399	131160	115475	54.26	44.6	0.41	0.73
	1	84368892	75322253	5855553	5997953	131260	115898	54.06	43.92	0.86	1.16
	4	84574176	75420681	5851330	6026833	131610	116016	51.47	40.58	3.87	4.08
	9	85090162	75742709	5926324	6235753	132356	116216	46.1	35.39	9.33	9.18
Dragon	0	84091807	74494490	5390368	5609145	131174	114816	48.51	49.34	2.04	0.11
	1	84125827	74529259	5376119	5570802	131260	114938	48.39	48.9	2.54	0.17
	4	84264808	74495558	5352952	5596834	131542	114855	46.55	47.8	4.95	0.7
	9	84684827	74788731	5437691	5808408	132106	114998	42.41	41.72	10.75	5.13
IBM04											
Feng Shui 1.5	0	235944107	211225865	19360473	19669252	254028	224679	58.26	40.26	0.61	0.87
	1	236210014	211757388	19252876	19506615	254466	225493	57.68	39.45	1.07	1.8
	4	236967174	212299011	19213858	19542885	255402	226088	54.29	37.09	3.68	4.95
	9	239798537	215010808	19681396	20406668	258174	228255	42.84	31.37	12.13	13.65
Dragon	0	235740458	207839911	18052738	18524017	254294	221155	47.39	49.29	2.22	1.1
	1	235968324	208207989	17962150	18389539	254666	221742	46.77	48.66	3.1	1.46
	4	236614283	208739886	17921496	18440312	255472	222308	43.9	46.39	6.12	3.59
	9	238931521	210571788	18346961	19180059	257682	223584	37.18	38.95	13.71	10.16

Table 2: Routing demand using an octilinear routing metric. Initially, most connections utilize the Manhattan routing layers. As routing cost changes, connections migrate to the non-Manhattan layers.

well as the complete set of MCNC benchmarks, and observe similar behavior; run time is roughly linear with the number of nets (as the time to construct a Steiner tree is independent of the size of the entire circuit).

Examination of large benchmarks reveals some "local" variation in routing demand, however. Thus, the appropriate application of our techniques is to optimize small portions of the design independently.

5. CONCLUSION AND FUTURE WORK

In this paper, we have presented a relatively simple method to adjust interconnect tree topologies to match routing resources. Our work combines prior methods for Manhattan Steiner trees[4], Non-Manhattan Steiner trees[10], and Preferred Direction Steiner trees[15], and then adjusts layer and via costs to obtain a satisfactory set of topologies and layer assignments.

Much of the current work in "routability driven" placement focuses on insertion of additional space to accomodate interconnect wires. Our work is complementary to this; analysis of routing demand on a per-layer basis can indicate if the placement should be stretched horizontally (to provide increased vertical capacity), vertically (for more horizontal capacity), or both.

We have integrated our layer balancing technique into a global router; preliminary experimental results are promising, and we will report results when this work is mature.

Acknowledgements: This work was supported in part by an IBM Faculty Partnership Award and funding from SRC under project 947.001.

6. REFERENCES

[1] The x initiative. http://www.xinitiative.org.

[2] S. Adya, M. C. YILDIZ, I. L. Markov, P. G. Villarubia, P. Parakh, and P. H. Madden. Benchmarking for large-scale placement and beyond. In *Proc. Int. Symp. on Physical Design*, 2003.

[3] C. Albrecht. Global routing by new approximation algorithms for multicommodity flow. *IEEE Trans. on Computer-Aided Design of Integrated Circuits andSystems*, 20(5):622–631, May 2001.

[4] M. Borah, R. M. Owens, and M. J. Irwin. An edge-based heuristic for Steiner routing. *IEEE Trans. on Computer-Aided Design of Integrated Circuits andSystems*, 13(12):1563–1568, December 1994.

[5] Andrew E. Caldwell, Andrew B. Kahng, and Igor L. Markov. Can recursive bisection alone produce routable placements? In *Proc. Design Automation Conf*, pages 477–482, 2000.

[6] H. Chen, B. Yao, F. Zhou, and C.-K. Cheng. The Y-Architecture: Yet another on-chip interconnect solution. In *Proc. Asia South Pacific Design Automation Conf.*, pages 09b–3, 2003.

[7] E. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269–271, 1959.

[8] A. B. Kahng, I. I. Mandoiu, and A. Z. Zelikovsky. Highly scalable algorithms for rectilinear and octilinear steiner trees. In *Proc. Asia South Pacific Design Automation Conf.*, pages 09b–1, 2003.

[9] A. B. Kahng and G. Robins. A new class of iterative Steiner tree heuristics with good performance. *IEEE Trans. on Computer-Aided Design of Integrated Circuits andSystems*, 11(7):893–902, July 1992.

[10] C.-K. Koh and P. H. Madden. Manhattan or non-manhattan? a study of alternative vlsi routing architectures. In *Proc. Great Lakes Symposium on VLSI*, pages 47–52, 2000.

[11] B. K. Nielsen, P. Winter, and M. Zachariasen. An exact algorithm for the uniformly-oriented Steiner tree problem. In *ESA '02*, 2002.

[12] D. Sylvester and K. Keutzer. A global wiring paradigm for deep submicron design. *IEEE Trans. on Computer-Aided Design of Integrated Circuits andSystems*, 19(2):242–252, February 2000.

[13] Maogang Wang, Xiaojian Yang, and Majid Sarrafzadeh. Dragon2000: Standard-cell placement tool for large industry circuits. In *Proc. Int. Conf. on Computer Aided Design*, pages 260–263, 2000.

[14] G. Y. Yan, A. Albrecht, G. H. F. Young, and C. K. Wong. The Steiner tree problem in orientation metrics. *J. Comp. Syst. Sci.*, 55:529–546, 1997.

[15] M. C. YILDIZ and P. H. Madden. Preferred direction multi-layer steiner trees. In *Proc. Great Lakes Symposium on VLSI*, pages 56–61, 2001.

[16] Mehmet Can YILDIZ and Patrick H. Madden. Improved cut sequences for partitioning based placement. In *Proc. Design Automation Conf*, pages 776–779, 2001.