

# **Multilayer Routing on Multichip Modules**

ECE 1387F  
CAD for Digital Circuit Synthesis and Layout

Professor Rose

Friday, December 24, 1999.

David Tam

(2332 words, not counting title page and reference section)

## Introduction

This paper will discuss the topic of multilayer routing on multichip modules (MCMs). In terms of circuit design flow, routing of an MCM occurs during the last step of the layout synthesis stage.

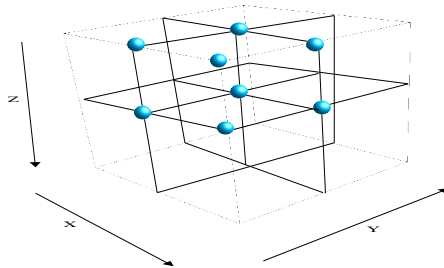
MCMs consist of a collection of IC cores (blocks) that are mounted on the top layer of a multilayer substrate and connected using the remaining lower layers [1]. Although the MCM routing problem may appear solvable using traditional printed circuit board (PCB) routers, MCMs contain many more routing layers and potential grid points [1]. Due to these differences, MCM routers have been developed by the research community.

To maximize speed on MCMs, the quality of a routing solution depends on the number of vias, wire length, the amount of cross-talk, and signal propagation delay [3]. Lower values of these characteristics are desired.

## Precise Problem Definition

Given:

- (1) A routing area in the form of a lattice structure:



- (2) A set of terminal points:

$T = \{ (x_0, y_0, 0), (x_1, y_1, 0), \dots, (x_t, y_t, 0) \mid (x, y, z) \text{ refers to co-ords in lattice struct,}$

$$0 \leq x \leq X, 0 \leq y \leq Y, 0 \leq z \leq Z \}$$

- (3) A set of nets:

$$N = \{ N_0, N_1, \dots, N_n \mid N = T, \forall_j \forall_k N_j \cap N_k = \phi \}$$

Find:

$\forall N_i$  find  $G(V,E)$  ,  
 where  $E = \{e((x_1,y_1,z_1), (x_2,y_2,z_2)) \mid e \text{ is an edge in lattice struct}\}$  ,  
 $N_i \subset V$  , and  $\forall_{j \neq i} \forall_{k \neq i} N_j \cup N_k \not\subset V$

Such that:

$\forall_i \forall_j G_i \cap G_j = \phi$  ,  
 while minimizing  $|V|$ ,  $|E|$ , and  $|e((x_1,y_1,z_i), (x_1,y_1,z_{i+1})) \in E|$

The routing path of each net  $N_i$  is given by the graph  $G_i(V,E)$ .  
 Vias are represented by  $e((x_1,y_1,z_i), (x_1,y_1,z_{i+1})) \in E$ , while total path length is represented by  $|E|$ . A routing layer is represented by  $(x,y,k)$ , where  $k$  is a constant and  $0 \leq k \leq Z$ .

## Approaches

The following approaches will be reviewed: (1) SEGRA [1], (2) an  $N \log N$  algorithm [3], and (3) MINOTAUR [2]. These approaches were chosen because they are the most recent developments. In most literature, approaches are often compared against a popular, pioneering algorithm called V4R [5].

### SEGRA

The SEGRA algorithm is described in [1] and is a simple, effective, greedy routing algorithm. It emphasizes speed and offers competitive quality in results. On average, it is 23 times faster than the closest competitor under the fast and effective category. When compared against the best solutions, SEGRA is on average 102 times faster. Routing time remains proportional to the size of the routing area. Multi-terminal nets are handled by decomposing them into two-terminal nets based on deriving a minimum spanning tree. Pairs of adjacent layers are sequentially selected from top to bottom and routing is attempted on each pair. This ordering is intended to route nets as close to the top layer as possible. The non-overlap routing convention is used, which specifies that within each pair of layers, one layer is designated to hold horizontal wire segments exclusively while the other layer is designated to hold vertical wire segments. Layers are conceptually viewed as consisting of Manhattan style grids, with terminal pins located on grid points.

At each layer, alternating horizontal and vertical sweeps are conducted from the outer edges towards the center. Two sweep lines begin at opposite outer edges and travel towards the center. Terminal points become candidates when intersected by the sweep line. During these sweeps, the trajectory of candidate terminal points are recorded. In constructing the trajectory, terminal points follow the path created by the sweep line and may move laterally to avoid potential obstacles (i.e. terminal points) or to become aligned with the target. If an obstacle cannot be avoided, the candidate point is removed and routing of the corresponding net is deferred to a lower layer. Lateral movement order is controlled by a six level priority scheme based on proximity to obstacles. Once the two sweep lines meet, paths of routable nets are revealed. After the horizontal and vertical sweeps are completed, unrouted terminals are extended to the next lower layer, unroutable nets are designated to the next lower layer pair and the process is repeated until all nets are routed.

### NlogN Algorithm

The NlogN algorithm is described in [3] and is a heuristic algorithm with time complexity of  $O(N \log N)$ , where  $N$  is the number of terminal points. It emphasizes minimal via count and total wire length. The algorithm initially follows the non-overlap routing convention and consists of three phases. Initially, a routing order is determined for the nets based on average path length. Shorter paths are selected ahead of longer paths. Next, nets are routed while minimizing usage of horizontal and vertical routing channels. Routing channels are defined as routing areas that are not directly below IC blocks. Finally, via minimization is attempted.

To determine the average path length of each net, a rectilinear steiner tree is constructed for each net that connects all relevant terminals. Edges of the steiner tree are mapped to horizontal and vertical routing channels, and routing areas under IC blocks are treated as obstacles in this stage. Besides physical distance, the average path of a net is also influenced by the number of terminal points. Nets containing many terminal points or large distances between terminal points are ordered near the beginning.

Routing is attempted on each layer pair, from top to bottom. A candidate net is selected from the order generated above. When nets are routed, several scenarios are handled explicitly. Routes that are completely contained under the routing area of an IC block are routed within the boundaries of the block and do not make use of horizontal or vertical routing channels. If a route must connect two IC blocks,

paths segments that do not make use of the routing channels are preferred. For instance, a preferred segment may enter and exit the interior routing space of a series of adjacent blocks in order to reach a destination terminal without making use of the routing channels. Similar to the SEGRA approach, unrouted terminal points are extended to the next lower level and the process is repeated until all nets are routed.

Once the initial routing is complete, several techniques are used to minimize via usage. Layer flipping may be used to swap horizontal and vertical wire paths. This local optimization technique avoids the inefficiencies of criss-crossing layers several times along a wire path. Track permutations are used to find an optimal placement of wire segments within a routing channel, which leads to a reduction in vias. Finally, the non-overlap routing convention is removed. Horizontal and vertical wire segments may migrate to the same layer if it will not interfere with other wire segments, thereby reducing via usage.

## MINOTAUR

The MINOTAUR router is described in [2] and is a hybrid technique that combines the technique of rip-up and reroute and iterative deletion. Global and detail routing are accomplished in separate steps. MINOTAUR focuses on the global routing aspects of the problem. Maximum congestion is the metric used in determined solution feasibility.

Although the rip-up and reroute method is known to be computational expensive, the developers believe it is necessary for an aggressive design. Each non-critical net is decomposed into a set of edges (two-terminal nets) using a minimum spanning tree algorithm. A maze router based on Lee's algorithm [4] is used to initially route the nets, followed by iterative rip-ups and reroutes of each edge. Congestion bounds, which influence path selection, are adjusted iteratively to allow the solution to converge to a desired final congestion value. This method only applies to non-critical nets, while critical nets are not disturbed. The congestion cost function used to influence path decision is selectable and affects the quality of the final results drastically. Three functions based on step-like, slope-like, and two-tier graphs are available.

In the iterative deletion stage, a redundant set of paths are first generated for each net and highly congested paths are iteratively removed. These paths are bounded by the bounding box paths possible between any two terminals. This method allows for quick estimation of global congestion.

Three permutations are possible to form the hybrid algorithm. The first permutation uses rip-up and reroute, adds the bounding box paths, and applies iterative deletion. The second permutation generates the bounding box paths, applies iterative deletion, and uses maze routing to add a new candidate path. The third permutation applies iterative deletion followed by rip-up and reroute.

## Discussion

These three approaches seem to have different primary goals in mind. SEGRA aims to achieve algorithmic speed while sacrificing a small amount of quality. The priority of NlogN is reversed. It aims to achieve quality and compromises on speed. Run times that were several minutes on SEGRA consumed several hours on traditional routers such as V4R [5] and SLICE [4]. Similarly, minutes in the traditional routers would only require seconds in SEGRA. In terms of

quality compromise, the range is between 3 - 18 %. However, under most circumstances, quality is actually improved. Under today's short time-to-market cycle and rapid advances in VLSI technology, the SEGRA approach appears to be very attractive [1]. SEGRA illustrates that greedy algorithms can work very effectively within the MCM routing domain.

The NlogN approach offers an average quality improvement of 12 % over SEGRA but the developers fail to give run time measurements. Attempting to compare time complexity was not possible either because very different components were used to determine complexity. Naturally, the complexity of the NlogN approach is  $O(N \log N)$ , where N is the number of terminal points. The time complexity of SEGRA is  $O((l+k)vn)$ , where l is the number of layers, k is a constant ranging from 1.10 to 11.23, v is the number of vertical routing channels, and n is the number of horizontal routing channels. The NlogN approach explicitly attempts to improve quality through path length and via reduction whereas the SEGRA approach attempts this implicitly. Under certain circumstances, this difference in philosophy may favour the NlogN approach.

The MINOTAUR algorithm approaches the MCM routing problem and solution from a very different perspective than SEGRA and NlogN. Since a global routing approach is taken, the results were presented in terms of maximum congestion values, which is more useful in determining the quality of the result that will be fed into a detail router. It appears the primary goal of MINOTAUR is to provide flexibility, modularity, and extensibility, rather than improving algorithmic speed. The flexibility of MINOTAUR removes the restriction of using adjacent layer pairs in routing horizontal and vertical wire segments. It does not restrict layers a route may use [2]. These qualities may allow MINOTAUR to easily adapt to new technological developments.

A potential disadvantage of SEGRA and the NlogN approach is that they do not handle critical nets or paths. This missing feature, which is addressed in MINOTAUR, may be quite important chip designers.

A potential source of inefficiency in SEGRA and MINOTAUR may be the conversion of multi-terminal nets into two-terminal equivalents. Such a partitioning may inhibit certain optimal solutions from being realized [3]. The NlogN algorithm does not suffer from this potential inefficiency since it considers multi-terminal nets in all calculations.

The NlogN algorithm may be too restrictive since it requires equal-sized, rectangular IC blocks, terminal points located along the edges of the block, and equal channel width between blocks. SEGRA provides a much more flexible environment and is not concerned with blocks at all. Instead, only terminals emanating from the IC blocks are considered.

The benefits of artificially imposing routing restrictions in the NlogN algorithm must be questioned. The algorithm attempts to avoid using horizontal or vertical channels as much as possible for fear of channel congestion. A significant amount of time is designated to this task. If this notion is abandoned completely, the resulting algorithm may appear to be similar to the SEGRA algorithm and run much faster. Channel congestion may never appear since paths would not be aware of any such concepts. More evenly distributed routing paths may result. Also, while trying to avoid the use of routing channels, the routing area directly under IC blocks may become overly congested and lead to unnecessary complications when connecting two terminals within a block.

The non-overlap routing technique used in SEGRA and NlogN, which restricts a routing layer to contain either strictly horizontal or vertical wire segments, appears to be very powerful. It can allow for a denser packing of nets into a given area. The addition of a wire segment causes minimal disturbance and interference since it is added in the same direction as other wire segments. The only disadvantage of this technique is that vias must be used to connect segments between the two layers. In the overlap routing technique, such as in a maze router, any change in initial direction of a path severely reduces the feasibility of placing future wire segments next to the first segment. The probability of future wires interfering with the directional change of the first segment is quite high.

## **Future**

Currently, SEGRA is not a performance driven MCM router. Modifying the priority scheme for net selection can accomplish this transformation [1]. As well, the greedy algorithm can always be tweaked for better performance [1]. Explicitly incorporating quality improvement goals into the algorithm may make this approach applicable to a wider user base. As well, handling of critical nets should be incorporated to provide a fundamental level of assurance.

Run time results for the NlogN algorithm need to be obtained

and published in order to compare algorithmic speed against other approaches. Handling of critical nets should also be incorporated. Measurements of congestion in routing channels versus areas under IC blocks should be obtained and studied to determine the effectiveness of congestion avoidance.

For the MINOTAUR algorithm, final results after a detail router run should to be obtained and published in order to compare the final quality of the solution. Several detail routers should be used and a sensitivity analysis should be performed.

## References

- [1] Y.-J. Cha, C.S. Rim, and K. Nakajima, "SEGRA: A Very Fast General Area Router for Multichip Modules," IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 18, May 1999, pp. 659-665.
- [2] J. Cong and P.H. Madden, "Performance Driven Multi-Layer General Area Router for PCB/MCM Designs," Proc. ACM Design Automation Conference, Jun. 1998, pp. 356-361.
- [3] S. Das, S.C. Nandy, and B.B. Bhattacharya, "High Performance MCM Routing: A New Approach," Proc. IEEE 12th Int. Conf. on VLSI Design, Jan. 1999, pp. 564-569.
- [4] K.-Y. Khoo and J. Cong, "A Fast Multilayer General Area Router for MCM Designs," IEEE Trans. on Circuits and Systems - II: Analog and Digital Signal Processing, vol. 39, Nov. 1992, pp. 841-851.
- [5] K.-Y. Khoo and J. Cong, "An Efficient Multilayer MCM Router Based on Four-Via Routing," IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 14, Oct. 1995, pp. 1277-1289.
- [6] T. Miyoshi, S. Wakabayashi, T. Koide, and N. Yoshida, "An MCM Routing Algorithm Considering Crosstalk," IEEE Int. Symp. On Circuits and Systems, May 1995, pp. 211-214.
- [7] D. Wang and E.S. Kuh, "A New Time-Driven Multilayer MCM/IC Routing Algorithm," IEEE Multi-Chip Module Conference, Feb. 1997, pp. 89-94.
- [8] T. Watanabe and T. Fujii, "A Hierarchical MCM Routing Using Four-Via Routing," Proc. IEEE Asia Pacific Conference on Circuits and Systems, Nov. 1996, pp. 389-392.