論文 86-23-4-18

A Study on Multilayer Routing Problem by CAD System

(CAD 시스템을 사용한 다층 Routing 문제에 관한 연구)

李 天 熙*

(Cheon Hee Yi)

要 約

본 연구는 다층 P. C. B. 를 위한 위상 기하학적인 내부연결 방법에 관한 것으로 고정된 관계의 시스템 내에서 절대적위치 보다는 내부연결 경로들의 상대적인 위치정함의 모형을 만들므로서 경로 형태상에 제한을 가하는 단순화를 피하는데 초점을 두었다.

Abstract

A topologically based interconnection routing of multilayer printed circuit boards has been proposed. This study focuses on modeling the relative positioning of the interconnect paths rather than absolute positioning within a fixed coordinate system, thereby avoiding simplifications that impose restriction on the path shapes.

I. Introduction

Research focused on developing algorithms to allow computers to layout multilayer routing system for integrated circuits and printed circuit boards. There are many similarities between layout problems for integrated circuits and printed circuit boards. These similarities suggest the use of similar layout approaches.

Integrated circuit connection is performed by diffusing impurities into the silicon surface or adding metalization patterns on top of the silicon to provide conducting paths between active elements. In this case the paths must usually go around or over, but not through, the components themselves, as the diffusion and the interconnection patterns realizing the element must be undisturbed.

In order to reduce size and enhance routability, several metalization layers, separated by layer of dielectric, are normally employed to provide additional interconnect resources. Transitions between metalization layers are performed by cutting holes in the dielectric and adding extra metalization on each layer to insure good electrical contact.

There are two objectives in the routing problem. One is to achieve routability. In other words, we want to make sure that all connections can be properly made. Integrated circuit and P.C.B are to be layed on each layer such that the circuits connection specifications are implemented by physical conductor wires. The other objective is the need of achieving high circuits density on the board, that is a large number of circuits components must be mounted on the board. Correspondingly this means more congested space for the integrated circuits and P.C.B.

A common problem shared by most conventional routers, such as maze running and

*正會員, 淸州大學校 電子工學科

(Dept. of Elec. Eng., Chongju Univ.)

接受日字 : 1985年 11月 24日

(* 本 論文은 83年度 IBRD 教育借款에 의한 教授國費海外派遣計劃에 의한 研究報告임.)

line probe routers, is that they lack in "topological fluidity",[1] that is, the capability to defer detailed wire patterns until all connections have been considered. The channel routers exhibit more topological fluidity. However, the traditional channel routing method is not flexible enough in permitting repositioning of wire generated in the channel assignment step.

II. Topological Routing

1. Forward Propagation

The routing problem is too complex to be dealt with as a whole; therefore it is necessary to partition the routing surface into substructures called "region." (see Fig. 2(b)).

The purpose of the forward propagation phase is to locate the lowest cost alternative for traversing regions; it is essentially a uniform-cost search technique.[2] Hulme and Wisniewski[3] give the steps for such a technique which they refer to as Dijkstra's algorithm with a heap sort as shown in Figure 2.

1. $f_s = 0$ and $f_i = \infty$, $i \neq s$.
2. U = [1,2, -------, n].
3. Create a heap in U, partially ordering nodes i according to their labels f_i.
4. While $U \neq 0$ do
5. $i = U_1$;
6. $U = U_1 = [\, i \,]$;
7. restore heap order in U;
8. for each $(i, j) \in A$ and $j \in U$ do
9. if $f_j > f_i + d_{i,j}$ then
10. $f_j = f_i + d_{i,j}$;
11. restore heap order in U.
12. stop

Fig. 2. Dijkstra's algorithm with a heap sort.

Note that the technique involves keeping a list (U) of all unexplored vertices. By imposing the restriction that all edge costs ($d_{i,j}$) are strictly positive, the vertex with the least total cost (f_i) is known to be the lowest possible cost to reach vertex i. The algorithm continues exploration until the lowest cost path to every vertex has been found.

When applying this technique to the region graph depicted in Figure 2(b), other considerations come to light. Finding a path to a region (vertex) on which a target borders may not indicate that a path to that target has been found. This is because previously defined interconnections within a region may block wires which enter on one particular edge from reaching some sections of the periphery. This problem can be solved by propagating from edge section to edge section which contains a target is reached has a path to that target been found.

Finally, by storing the edge sections in an exploration queue which is ordered by cost, the steps which restore the heap order can be eliminated. The topological router forward propagation algorithm is presented in Figure 2(a).(Q: a list of edge section ordered by cost T: target set)

1. Q = [S]
2. While $Q \neq 0$ and not $i \in T$
3. i = Q
4. Q = Q − [i] ; U = U − [i]
5. for each $(i,j) \in A$ and $j \in U$ do
6. if $(i \in T)$ Q = Q + [j]
7. Stop.

Fig. 2(a). Topological router forward propagation algorithm.

The determination of reachable edge sections in step 5 requires a special operation involving the planarity list. Given a pair of crossover records belonging to a single boundaries which can be reached without crossing over any other wiring.

2. Backtrack

One the uniform cost forward propagation technique has located the best path, the task remains to trace the definition of this path back to its source. This is designated as backtracking. The tree created by forward propagation contains pointers from each vertex to the one which preceded it in the shortest path to that point. The task of backtracking is to walk back up the exploration tree along the shortest path, using these pointers and

perform updates of the crossover lists of the edges to include the crossovers represented by this route.

Physically, the crossover list associated with each edge is a doubly linked list. When forward propagation passes between two different entries on a single edge, the insertion point for the new crossover during backtrack is obviously between those two crossovers. This also requires that the allowable range or float associated with neighboring crossovers on the list be reduced to provide room for the new entry.

Selection of the insertion point for the new entry sometimes involves an arbitrary decision that is the crossovers between which forward propagation passes sometimes are not adjacent.

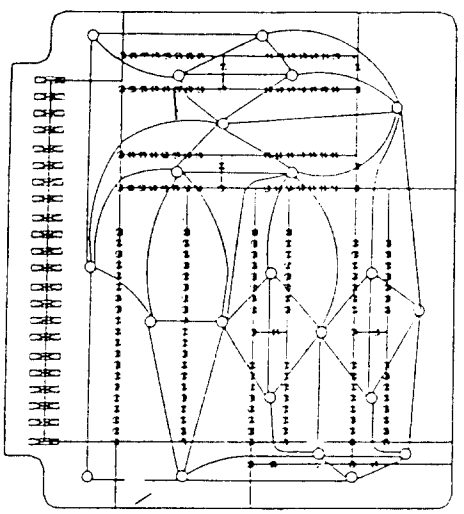


Fig. 2(b). A printed circuit board with the Associated region graph

3. Complexity Analysis

The dominant factor in execution time is the path exploration forward propagation. backtrack simply traces back one of the paths from the leaf nodes to the root of the exploration tree. Forward propagation, on the other hand, will have explored and built the entire tree. The forward propagation phase will itself be dominated by the determination of reachable edge sections in step 5.(see Fig. 2(a))

For each region explored, the planarity list is scanned to determine reachable edge sections.

The number of crossovers in a given planarity list increases with each connection routed. Therefore, exploration time will be greater for connections done late in the initial route stage and all during iterative improvement than it is for connections made earlier on. This is because model size is a function of the number of connections defined and not strictly a function of the surface area being modeled.

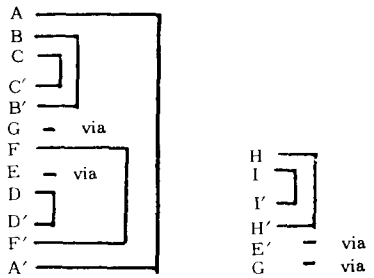
The number of edges sections explored is highly dependent on the cost factors employed. Since forward propagation is a uniform cost search, costs can be selected so as to make it approximate either a breadth first or depth first search[4] in complexity.

III. Multi-layer Contour Routing

There are two options for processing vias during topological routing which are the finding the minimum cost path in a weighted graph for each necessary connection and planarity consideration at the vertices (regions).

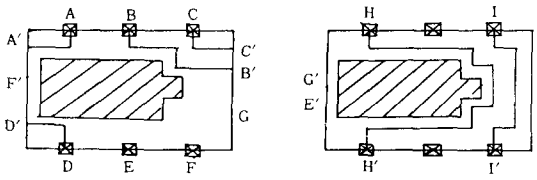
This provides a large number of potential via sites and therefore a high degree of flexibility, but the model used provides inexact information about physical space available for the vias during topological routing. The second option restricts via sites to lie along the bounding edges. This reduces the number of available via sites, but geometric information needed to determine if a particular via will fit at a certain location is known precisely. For the contour routing approach to coordinate assignment option two is the easier to implement. A via on a boundary is processed like a connection point and the procedure presented in Dorau's paper.[5]

The solution to this problem is to use the planarity list as before to order connections by their adjacency in the list. A planarity list is an ordered list of crossovers around the periphery of a region. A planarity list is formed by concatenation of the crossover lists for the edges of the region, ordered in a clockwise direction around the region periphery. However, contour routing is performed as far as possible on each layer of the region before any via version having two vias, the planarity lists are shown in Figure 3 and the initial contour routing in Figure 3(a).



a) Layer 1. b) Layer 2.

Fig. 3. Original planarity lists for two-via problem.



a) Layer 1. b) Layer 2.

Fig. 3(a). Initial contour routing for two-via problem.

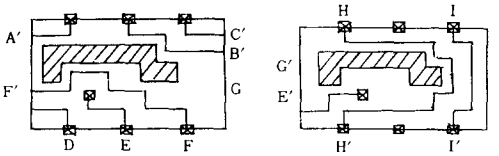
Notice that in this case all of the contour routing cannot be completed prior to via site slection because no more on-layer connections are adjacent in the planarity list (see Fig. 3(b)). If the via site is not carefully selected, it could preclude an otherwise viable solution. Looking at layer one, connection E's via should be placed near the bottom edge. According to the topology of layer two it should be near the left edge. A similar conflict exists for connection G. A heuristic decision must be made here.

It is instructive to complete the example by manually selecting a via site. Choose connection E's via site to be between its two connection points on the outer contour of the allowable area. This places it as far out of the way of subsequent connections as possible. The remaining single layer connection (F) becomes adjacent when E is removed from the planarity lists. The contour routing is shown in Figure 3(c).

In this case there is available area for the connection G via. Notice that there are sites which, if selected for connection E's via, would

a) Layer 1. b) Layer 2.

Fig. 3(b). Compressed planarity lists for two-via problem.



a) Layer 1. b) Layer 2.

Fig. 3(c). Contour routing for two-via problem.

have rendered G unroutable. One such site is one grid above the chosen site for connection E's via.

The contour router performs the transformation from the topological route model into the geometrical coordinate realm. The technique is one which, in the absence of layer transitions, can guarantee a soultion in which all connections within the given region are routed, if such a solution exists. Two or more layer transitions within a single region require heuristic decisions to be made. The via problem in contour routing emphasizes the need for careful tuning of the topological route phase to reduce layer transitions as much as possible.

Coordinate assignment is third and final phase of the topological interconnection routing system. The first phase, region generation, identifies open areas of the board and designates them as "regions" in the topological model. The second phase, topological routing, specifies relative positioning of the interconnection within the model. And finally, coordinate assignment assigns detailed wiring paths for the interconnection wiring.

IV. Experimental Results

1. Design of Experiments

A experiments have been performed to both determine the feasibility of the topological routing technique as a whole and to explore

the effects of various costing strategies on the topological exploration. A description of the experimentation performed using these routing techniques is presented along with the conclusions.

Several sample printed circuit boards have been identified to serve as test cases for experimentation. They span a fairly large range of board sizes and component densities (the number of components per square inch of surface area) in order to demonstrate performance of the system on a wide range of problem sizes and difficulties. This variety will also serve to illuminate any variation in performance or turning characteristics that are a function of problem size of complexity.

The individual board characteristics are summarized in table 4. Component density is reflected in the columns labeled "square inches per pin" and "square inches per IC equivalent." The "IC equivalent" measure is determined by simply assuming an average to provide an analogy to a circuit board populated with fourteen and sixteen pin dual-in-line packages. The test cases in actuality consist of dual-in-line packages of various sizes along with the associated decoupling capacitors, edge connectors and test points.

The test cases are derived from several different sources. Test case A contains miscellaneous logic and is one of the examples used in Brady's Dissertation [6] to demonstrate the automatic generation of logic diagrams. Test case B is an example designed specifically for this research. It is intended to represent a microprocessor and some associated logic, although many of the interconnection requirements were specified randomly. Because it has regions of various sizes and components in different orientations it demonstrates the flexibility of the system well. Also, it is of size and density which make it suitable for illustrations. It consists of 6200 lines of FORTRAN code (including comments) and runs on a VAX 11/780 computer. Parameters used in the cost calculations will be accepted as inputs to be program to permit convenient tuning for layer transition elimination experiment.

Test cases C and D [7,8] are printed circuit boards containing prototype circuit. Test case E is a switching network from the texas reconfigurable array computer project (TRAC). [9,10]

Table 4. Test board characterization

Test Case	Number of Nets	Number of Connections	Number of Pins (used / total)	Dimensions (length x width / layers)	Square Inches per Pin	Square Inches per IC equivalent
A	45	83	128/136	4×6 / 2	0.42	2.65
B	65	140	205/222	4×4.5 / 2	0.17	1.22
C	96	363	459/619	7×7.25 / 2	0.08	1.22
D	162	471	633/761	7×7.25 / 2	0.08	1.00
E	255	720	974/1149	6×15 / 2	0.067	1.17

2. *Layer Transition Elimination*

A layer transition is used when a connection between the two points on the other layer is not possible due to the wiring on each layer, since the connections are movable within the region by extending the wires. Physically, there may not always be room for such adjustments in the positions of the connections, but in the absence of wire width and spacing restrictions such a solution always exists.

Since layer transitions are known to be a source of potential problems in the topological approach, it is important to determine the extent to which they can be totally eliminated. In this experiment, the cost for a layer transition is set to a prohibitively large value, and topological routing is performed as far as possible. Note that some connections will fail to be made and the number of such connections is the primary dependent variable here. Parameter settings are as follows:

Layer transition cost = Infinity

Net ordering = shortest to longest perimeter

The results are summarized in Table 4(a). The topological route results for test case B are depicted in Figure 4 and 4(a). Careful study of these pictures reveal numerous instance where the router was able to weave in and out of rows of pins as required. Also, the wiring seems to be distributed somewhat uniformly between the two layers.

V. Conclusions

The feasibility of the topological routing technique has been demonstrated. The number

Table 4(a). Layer transition elimination results.

Test Case	CPU Time	Number of Vias	Failed Connections	Failed Nets	Completion Rate	Esitmated Wire Length
A	2:07	0	1	1	98.8%	242.0
B	6:26	0	14	12	90.0%	301.7
C	48:26	0	13	10	96.4%	819.0
D	40:50	0	107	50	77.3%	1186.3
E	2:17:17	0	103	65	85.7%	2268.1

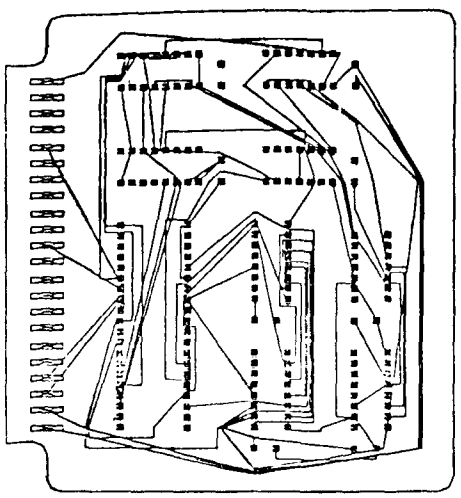


Fig. 4. Layer transition elimination experiment, layer 1 of test case B.

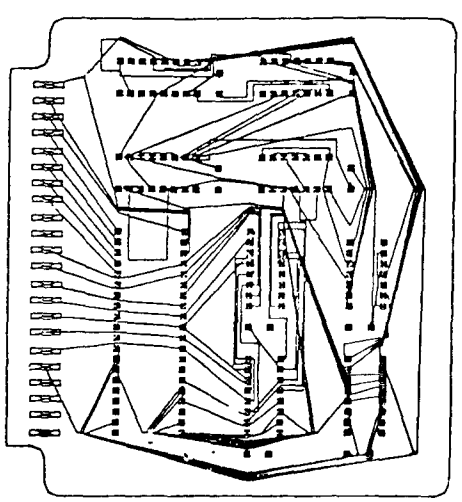


Fig. 4(a). Layer transition elimination experiment, layer 2 of test case B.

of vias per connection is reduced below that of conventional routers. It has been shown that associating a preferred routing direction with individual layers results in little improvement. This indicates that such techniques reduce the fluidity which is the primary advantage of this routing system. Finally, computational complexity appears to be $O(n^2)$ where n is the number of connections attemped.

Areas for future work is both approaches to a via handling on the bounding edges and within the regions should be explored further. This includes their handling in both the topological routing phase and coordinate assignment phase. Experimentation to determine the degree of the success of the heuristic techniques on practical high density problems would be of great interest.

Acknowledgements

The author is indebted to Dr. John Mitchem and Dr. Brad Jackson of San Jose State University for their helpful discussions during the course of this work. He also would like to specifically acknowledge The Ministry of education and Cheong-Ju University for their financial support of this work.

Reference

[1] Doreau, Michel T., and Luther C. Abel, *A Topologically Based Non-Minimum Disttance Routing Algorithm.* Proc. 15th Design Automation Conference, pp. 92-99, 1979.

[2] Barr, A. and Feigenbaum, E.A., The Handbook of Artificial Intelligence, vol. 1, William Kaufmann, Inc., 1981.

[3] Hulme, B.L. and Wisniewski, J.A., *A Camparison of Shortest Path Algorithms Applied to Sparse Graphs.* Sandia Laboratories, August 1978.

[4] Rebert Tarjan, "Depth-first Search and Linear graph Algorithms," *SIAM J. Computer*, vol. 1, no. 2, June 1972.

[5] Doreau M.T., Koziol, P., *TWIGY-A Topological Algorithm Based Routing System.* 18th Design Automation Conf. Proc. pp. 746-755, 1981.

[6] Brady, H.N., *Automatic Generation of Schematic Diagrams.* Ph. D Dissertation at

the University of Texas at Austin, 1982.

[7] L.M. Jay, *Algorithm and Models for a Topologically Based Interconnection Routing System.* Ph. D Dissertation, The University of Texas at Austin, 1983.

[8] T,R. Yuk-Pan, *Automated Routing Techniques for High-Density Multilayer Printed Circuit Boards.* Ph. D Dissertation, The University of Texas at Austin, 1982.

[9] Premkumar, U.V., Kapur, R., Mlck, M., Lipouski, G.J. and Horne, P., *Design and Implementation of The Banyen Interconnection Network in TRAC.* AFIPS Conference Proceedings NCC, pp. 643-653, 1980.

[10] Tsui, R. and Smith, R.J. *A High-density Multilayer PCB Router Based on Necessary and Sufficient Conditions for Single Row Routing.* 18th Design Automation Conference Proceedings, pp. 372-381, June, 1981.