Via Minimization in VLSI Routing with Movable Terminals

JITENDER S. DEOGUN AND
BHARGAB B. BHATTACHARYA

Abstract—**In this paper we develop a unified approach for solving the general problem of minimizing the number of "via holes" in a two-layer VLSI channel and switch-box routing environment with movable terminals. All horizontal segments of the nets are assumed to be in one layer, and the vertical segments in the other layer. Each net can have multiple terminals. Three different models are considered: (i) two-row channel routing, (ii) three-sided switch-box routing, and (iii) four-sided switch-box routing.**

To solve the via minimization problem, we introduce the concept of a *maximum parallel set* of edges in a bipartite graph. This leads to a unified graph-theoretic approach for solving the via minimization problem for all three models considered. The complexity of the proposed algorithm is $O(N \log N)$, in all three cases, where N is the number of pairs of terminals to be connected.

Keywords—**Design automation, VLSI channel and switch-box routing, via-minimization, permutation graphs.**

I. Introduction

The channel routing problem is an important issue in the layout design of LSI/VLSI circuits [1], [5]. In particular, the problem of local channel routing, commonly known as *permutation channel routing*, has received considerable attention in recent years. The channel routing problem has been investigated under different models and assumptions [1], [3], [5]–[8]. The different criteria of minimality considered include the number of horizontal tracks, the number of vertical tracks, and the number of "vias." The problem of minimizing the number of horizontal tracks when the terminals have fixed positions on two sides of a channel has been shown to be *NP*-complete both with and without doglegs [6], [7]. However, a number of efficient heuristic algorithms are known to exist [1], [5]. Similarly, the problem of minimum-via topological routing is also known to be *NP*-complete [2]. Under the assumption of movable terminals the problem of minimizing the number of horizontal tracks and that of minimizing the number of vias can be solved polynomially for the case of permutation channel routing [3].

In this paper we develop a unified approach for solving the general problem of minimizing the number of via holes in a two-layer VLSI channel and switch-box routing with movable terminals. In two-layer routing, the horizontal segments of the nets are assumed to be in one layer, and the vertical segments in the other. It may be noted that if a net has all its terminals on just one side of the channel or switch box, then the number of vias for this net is fixed. Moreover, such a net does not affect the via minimization of other nets, and thus can be inserted after the solution for remaining nets has been found without affecting the solution. We, therefore, exclude such nets from the discussion that follows. We consider the problem of via minimization with *multiple-terminal nets*, from three different perspectives:

(i) *Two-row channel routing:* This model assumes two horizontal rows of terminals which can be *moved freely*, keeping their relative order unchanged.

(ii) *Three-sided switch-box routing:* In this model, two horizontal rows of *movable terminals* are considered, but in addition, a vertical column of fixed terminals is also assumed to be present, thus forming a three-sided switch box.

(iii) *Four-sided switch-box routing:* A four-sided switch box comprises two horizontal rows of movable terminals as well as two vertical columns of *movable terminals*. The terminals are assumed to be movable on respective sides without changing their relative order.

An algorithm suitable only for the simple case where each net has exactly two terminals under model (i) and has complexity $O(N^2)$ was presented in [3]. In contrast, in this paper we present an algorithm that is applicable to all three models and is of complexity $O(N \log N)$, where N denotes the number of pairs of terminals to be connected. A discussion of the concepts used in this paper and the details of the model for movable terminals can be found in [3].

II. The Problem Formulation

Let us first consider the simple problem of via minimization in permutation channel routing, in which every net connects exactly two terminals, one at the top row and the other at the bottom row of the channel. In such a routing problem, if we label the terminals of the top row $a_1, a_2, \cdots, a_N$ and those of the bottom $b_1, b_2, \cdots, b_N$, then the layout problem clearly corresponds to a permutation

$$\Pi: \{1, 2, \cdots, N\} \rightarrow \{1, 2, \cdots, N\}$$

where the net i imparts a connection between terminals a_i and $b_{\Pi(i)}$.

It has been shown that the problem of minimizing the number of via holes in two-layer routing with movable terminals is equivalent to the problem of maximizing the number of abutments [3]. Clearly, if two nets $\{a_i, b_{\Pi(i)}\}$ and $\{a_j, b_{\Pi(j)}\}$ are intersecting, then both cannot be wired simultaneously using abutments. Therefore, minimizing the via-holes is equivalent to finding a maximum independent set of vertices in $G(\Pi)$, where $G(\Pi)$ is the permutation graph associated with the permutation Π [4], [12]. It is known that the problem of finding the maximum independent set in a permutation graph is equivalent to the problem of finding the longest up-sequence in the corresponding Π^{-1} [4]. Thus the via minimization problem can be solved by finding the longest up-sequence in Π^{-1}. The longest up-sequence problem can be solved in $O(N \log N)$ time [9]. Various improvements of these algorithms have also been suggested [10].

In this paper we consider the general two-row channel routing problem where a net might consist of multiple terminals as shown in Fig. 1. We will also consider three-sided and four-sided switch-box routing problems.

III. A. Unified Approach

To solve the general problem of via minimization, we introduce the concept of a *maximum parallel set* (MPS) of edges in a bipartite graph. Let $G = (X_1, X_2, E)$ denote a bipartite graph. Consider a

Manuscript received April 21, 1987; revised February 10, 1988, June 7, 1988, and January 26, 1989. An earlier version of this paper appeared in the Proceedings of the IEEE 2nd International Conference on Computers and Applications, Beijing, China, June 24–26, 1987. The review of this paper was arranged by Associate Editor R. H. J. M. Otten.

The authors are with the Department of Computer Science, University of Nebraska at Lincoln, Lincoln, NE 68588.

IEEE Log Number 8927534.

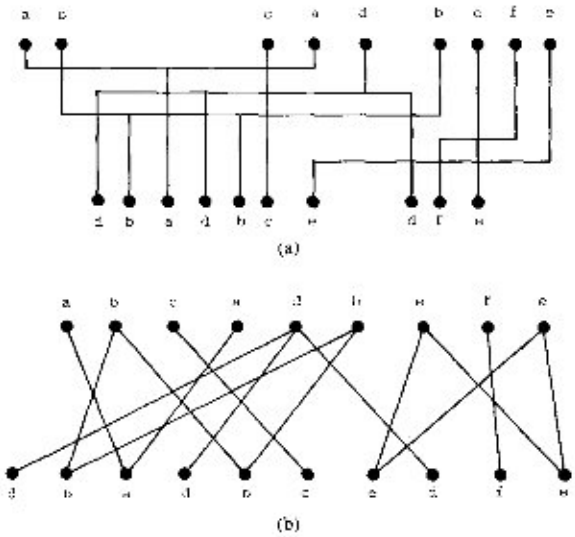

Fig. 1. (a) A two-row multiterminal routing problem. (b) Its bipartite graph.

geometrical embedding of G such that all vertices of X_1 and X_2 lie on two straight lines, respectively, and an edge (u, v), $u \in X_1$, $v \in X_2$ is drawn as a straight line between the vertices u and v. Two edges e_1 and e_2 in E are said to be parallel if e_1 and e_2 do not intersect geometrically in the given embedding and if e_1 and e_2 do not share a common vertex either in X_1 or in X_2. A subset $D \subseteq E$ of edges is called a *parallel set* if every pair (e_i, e_j), $e_i, e_j \in D$, is parallel. A parallel set is referred to as non-cross matching in [11]. A parallel set D is said to be a *maximum parallel set* if there is no other parallel set D' such that $|D'| > |D|$.

It is shown that the problem of minimizing via holes under the three models described can be transformed into one of finding maximum parallel sets in bipartite graphs describing the net connectivity.

A. Two-Row Channel Routing

Consider a two-row multiterminal routing problem as shown in Fig. 1(a), in which the number of vias is to be minimized. The connectivity can be represented by a bipartite graph G as shown in Fig. 1(b). It may be noted that for a multiterminal net (i.e., having three or more terminals), none of the connections of which is an abutment, selecting one vertical abutment reduces the number of via holes by one. However, realization of a two-terminal net by an abutment lowers the count by 2. To incorporate this, we consider a *weighted bipartite graph* G_w such that all the edges in G receive a weight equal to either 1 or 2. In G an edge (u, v), $u \in x_1$ and $v \in X_2$, is said to be *isolated* if $\text{degree}_G(u) = 1$, and $\text{degree}_G(v) = 1$. The graph G_w is constructed by associating a weight 2 to all the isolated edges and the weight 1 to all other edges.

A *maximum weighted parallel set* of edges in G_w is defined to be a parallel set of edges such that their cumulative weight is maximized. We can now state the following important result

Theorem 1: The set of abutments which minimizes the number of via holes in a two-row multiterminal channel routing with movable terminals is given by the maximum weighted parallel set of edges in the bipartite graph G_w describing the net connectivity.

Proof: Immediate from the above discussion.

The problem of finding the maximum weighted parallel set in a bipartite graph can be transformed to the problem of finding a maximum independent set of an equivalent permutation graph as described below. Given a weighted bipartite graph, construct a permutation diagram as follows:

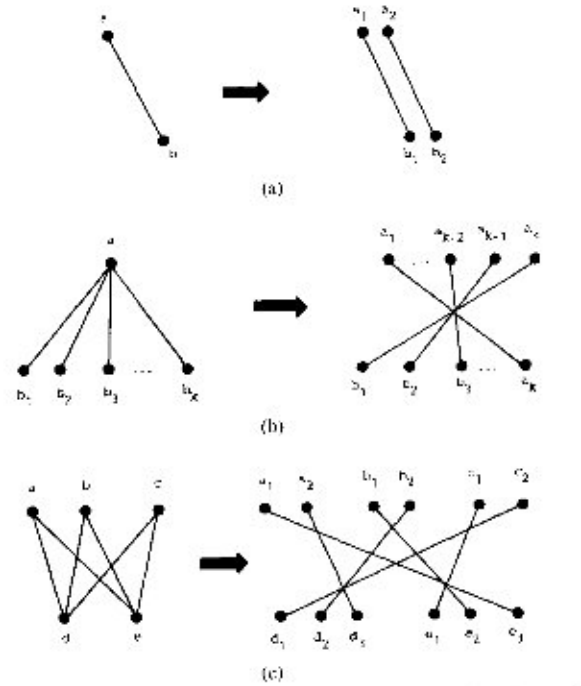

Fig. 2. Construction of permutation diagram from weighted bipartite graph.

(i) For each isolated edge in G_w, create two consecutive parallel edges as shown in Fig. 2(a).
(ii) For each multiterminal net, say a, create a consecutive set of mutually intersecting edges $a_1, a_2, \cdots, a_k$ as shown in Fig. 2(b) and (c).

The transformation is straightforward and can be affected in $O(N)$ time, where N is a number of pairs of terminals to be connected. It should be noted that in the above transformation when a vertex, say a, is split into several vertices $a_1, a_2, \cdots, a_k$, as shown in Fig. 2(b) and (c), all the split vertices intersect with each other and appear consecutively in equivalent permutation diagram.

Lemma 1: The maximum weighted parallel set of edges in G_w can be computed by finding the maximum independent set of vertices in the permutation graph $G(\Pi)$ defined by the permutation diagram constructed from G_w.

Proof: From the construction of permutation diagram it follows that if two edges in G_w intersect geometrically or have a common vertex, then the corresponding lines in permutation diagrams intersect geometrically. Moreover, an edge that has weight 2 in G_w contributes two consecutive parallel lines in the permutation diagram. Hence the lemma.

Example: Consider the via-minimization problem for the multiterminal two-row routing problem shown in Fig. 1(a), whose graph is shown in Fig. 1(b). The equivalent permutation diagram after transformation is shown in Fig. 3. The final solution, shown in Fig. 4, contains 11 vias, which is indeed the minimum.

Theorem 2: The worst-case complexity of the via-minimization algorithm in a multiterminal two-row routing problem is $O(N \log N)$, where N is the number of pairs of terminals to be connected.

Proof: The transformation of the weighted bipartite graph to a permutation diagram can be affected in $O(N)$ time, because the number of vertices in the equivalent permutation graph can be at most $2N$. Since the maximum independent set problem can be solved in $O(N \log N)$ time in a permutation graph [4], [12], the proof follows.

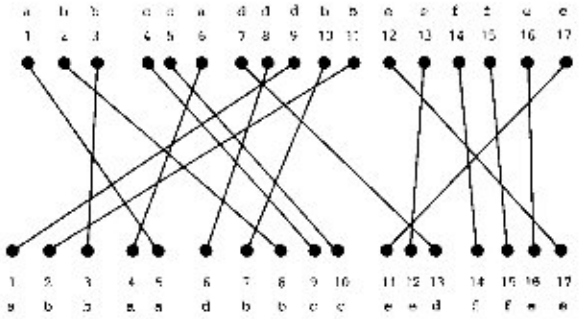

Fig. 3. Permutation diagram for the problem in Fig. 1.

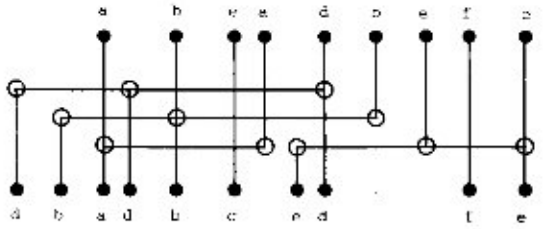

Fig. 4. Optimal solution to the problem in Fig. 1.

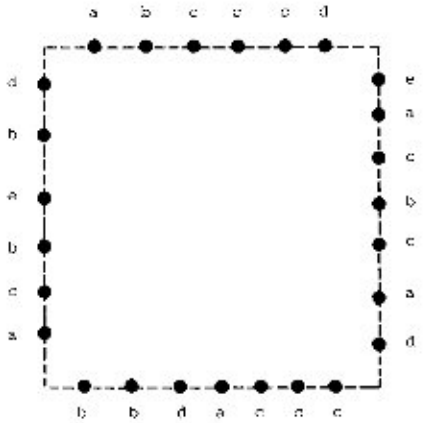

Fig. 5. Four-way switch-box routing problem.

B. Three-Sided and Four-Sided Switch-Box Routing Problem

The problem of via minimization in a switch-box environment with movable terminals can be managed by solving at most two independent MPS problems (one in the horizontal direction, and the other in the vertical direction) and then properly merging the results of these two MPS problems. It may be noted that the independent solution of two MPS problems may not always lead to the optimal solution in the switch-box environment. We will illustrate the method for a four-sided switch-box problem. The "three-sided" problem is similar.

Four-Sided Switch-Box Routing: Consider the routing problem shown in Fig. 5 (terminals with same labels should be mutually connected). We partition the problem into two equivalent weighted MPS problems for the two bipartite graphs corresponding to the vertical and horizontal connectivities (see Fig. 6(a) and (b)). Note that the number of vias necessary to establish cross-connectivity among horizontal (vertical) and vertical (horizontal) terminals is independent of their placement. The equivalent permutation diagrams after transformation are shown in Fig. 7(a) and (b).

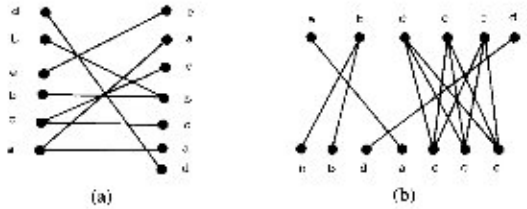

Fig. 6. Bipartite graphs for vertical and horizontal connectivities. (a) Horizontal bipartite graph. (b) Vertical bipartite graph.

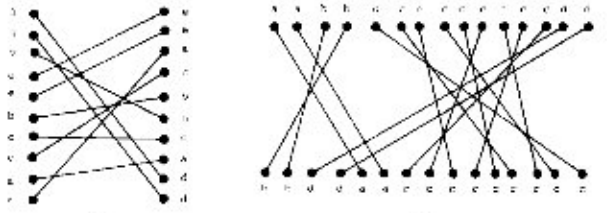

Fig. 7. Permutation diagrams after transformation. (a) Horizontal permutation diagram. (b) Vertical permutation diagram.

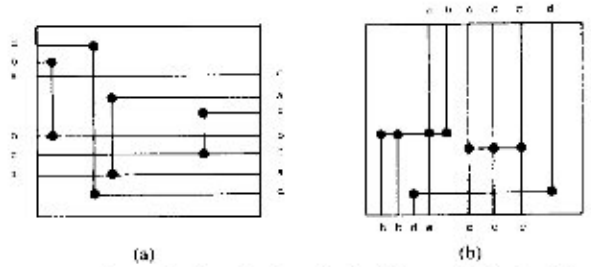

Fig. 8. Solutions to horizontal and vertical problems. (a) Horizontal solution. (b) Vertical solution.

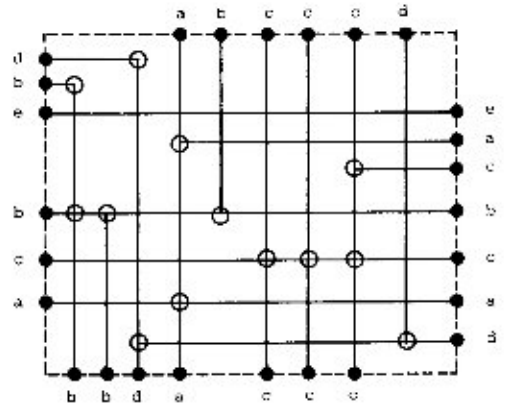

Fig. 9. Optimal solution of four-way switch-box routing problem.

Once the maximum independent sets in these two graphs are found, we can determine the maximum set of abutments in both directions. In developing the layouts for horizontal (vertical) connectivities, the vertical (horizontal) tracks should be aligned with vertical (horizontal) tracks required for vertical (horizontal) connectivities. The layouts for horizontal and vertical connectivities are shown in Fig. 8(a) and (b) respectively. The horizontal and vertical layouts are then superimposed, and in order to connect

identical termir s in horizontal and vertical directions necessary tracks and vias c inserted and the final solution is obtained. The final routing is own in Fig. 9, which minimizes the number of vias (= 13).

Corollary The complexity of the via-minimization algorithms in a f r-way switch-box routing problem is $O(N \log N)$.

Proof: ollows immediately from Theorem 2.

IV. Conclusion

We d cribe a unified graph-theoretic approach for minimizing the nur er of via holes for a generalized two-layer routing problem wher erminals are freely movable without changing the relative ord The algorithm is equally applicable two two-row channel routing and to three- and four-sided switch-box routing models.

References

[1] T. Yoshimura and E. S. Kuh, "Efficient algorithms for channel routing," *IEEE Trans. Computer-Aided Design*, vol. CAD-1, pp. 25–35, Jan. 1982.

[2] C. P. Hsu, "Minimum-via topological routing," *IEEE Trans. Computer-Aided Design*, vol. CAD-2, pp. 235–246, Oct. 1983.

[3] L. S. Gopal, D. Coppersmith, and C. K. Wong, "Optimal wiring of movable terminals," *IEEE Trans. Comput.*, vol. C-32, pp. 845–850, 1983.

[4] M. C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*. New York: Academic, 1980.

[5] J. Soukup, "Circuit layout," *Proc. IEEE*, vol. 69, pp. 1281–1304, Oct. 1981.

[6] S. Sahni and A. Bhatt, "The complexity of design automation problems," in *Proc. ACM-IEEE 17th Design Automat. Conf.*, June 1980, pp. 402–411.

[7] T. G. Szymanski, "Dogleg channel routing is NP-complete," *IEEE Trans. Computer-Aided Design*, vol. CAD-4, pp. 31–41, Jan. 1985.

[8] Y. Kajitani, "On via hole minimization of routing on a two-layer board," in *Proc. ICCC*, 1980, pp. 295–298.

[9] D. Gries, *The Science of Programming*. Berlin, Germany: Springer-Verlag, 1980.

[10] P. Widmayer and C. K. Wong, "An optimal algorithm for the maximum alignment of terminals," *Information Process. Lett.*, vol. 20, pp. 75–82, Feb. 1985.

[11] Y. Kajitani and T. Takahashi, "The noncross matching and applications to the 3 sided switch box routing in VLSI layout design," in *Proc. ISCAS*, May 1986, pp. 776–779.

[12] L. Lovasz, "A characterization of perfect graphs," *J. Comb. Theory*, vol. 13B, pp. 95–98, 1972.

On the Design and Implementation of a Wafer Yield Editor

JOSE PINEDA DE GYVEZ AND J. A. G. JESS

***Abstract*—An interactive environment for the analysis of yield information required on modern integrated circuit manufacturing lines is presented. The system estimates wafer yields and wafer yield variations, quantifies regional yield variations within wafers, identifies clusters in wafers and or in lots, and is also able to predict wafer yields via simple simulation tools. An analysis approach based on site yields makes the system also independent of the product and of the technology. The analysis technique investigates the effect of both correlated and uncorrelated sources of yield loss. The statistical information obtained can be used to study the changes in the technological process. Graphical displays in the form of wafer maps are used to represent the spatial distribution of dice in the wafer. Facilities such as radial and angular distribution analyses, among others, are provided to examine data, and hypothetical wafer maps are created to visualize and predict simulated wafer yields.**

Manuscript received June 16, 1988; revised October 31, 1988 and February 15, 1989. This work was supported by the Dutch Department of Economic Affairs under the IOPIC program under project IC-EEL46018. The review of this paper was arranged by Associate Editor M. R. Lightner.

The authors are with the Department of Electrical Engineering, Eindhoven University of Technology, 5600 MB, Eindhoven, The Netherlands.

IEEE Log Number 8927945.

I. Introduction

The yield associated with individual process steps like etching, metallization, etc., as well as the spatial distribution of random and systematic sources of yield loss [1]–[3], [19] have a different impact on each of the IC's of a wafer, such that yield variations and even sometimes severe product yield losses arise.

Such yield variations can be investigated by examining batches of wafers in order to correlate non-functional circuits and to find their contributions to yield loss. Interaction with these yield variations can be aided by analyzing the wafer maps where functional, nonfunctional and partially functional regions can easily be observed. Considering this, a tool is necessary to manage the data coming from the manufacturing lines and to condense it in useful information for whom yield prediction and estimation are very important issues for the IC design and process development.

Such a CAD tool should be able to analyze the yield variations, to quantify them, and to allow the interpretation of data in several forms as to draw conclusions about the problems. Furthermore, it should not only analyze data but also simulate the effects of density variations in a single wafer and between wafers, as to be able to predict yield. For such a system, *technology independency* and *product independency* are two mandatory properties. The former one concerns the analysis of data coming from GaAs, MOS, BIPOLAR, etc., technologies, and, for any given wafer diameter and dice configuration. The latter property means that the system should interpret data such as distribution of defects, process parameters, test structures, memories, etc.

This paper describes the *Wafer Yield Editor* ALWAYS (ANalyzer of WAfer YieldS). ALWAYS is a user friendly interactive environment created to be used as an integral and systematic tool for wafer yield analysis. The system provides means for estimation and prediction of yield. It uses graphical representations in the form of wafer maps, curves, and charts for user interface. Facilities such as to create wafer frames and dice of different dimensions, as well as several miscellaneous tools such as hardcopy, overlapping of extracted wafer maps, etc., are also provided.

II. Database and Wafer Characteristics Description

The starting data are a set of wafer maps of working and nonworking dice of individual wafers produced from the process of interest.

A set of wafers is defined as a *lot* and a set of *lots* as a *project*. This classification allows to have the information in a hierarchical style. Hence, the database description follows a tree structure where the parent is the product itself, the children represent each one of the lots of the product, and the grandchildren represent the individual wafers for each lot. Each individual wafer contains the input data of the yield editor, and it consists of all the die positions and their status, ON, or OFF.

Since product independency is one property of the editor, the data supplied may concern linewidths, resistivities, oxide thicknesses, etc., or, defect distributions, distribution of opens and shorts in different layers, distribution of good and bad chips, etc. This flexibility allows to set the status of each die according to the convenience of the analysis to be performed. For instance, consider that a project consists of a memory chip where the status of each die can be set as ON for functional chips and as OFF for defective ones. In this case any analysis performed will reveal the yield of the memory chip. On the other hand, if the project represents a