

プリント基板レイアウト設計における非平面接続数の 極小化手法

水野健二† 渡邊敏正††

†広島大学大学院工学研究科 情報工学専攻

††広島大学工学部 第二類 回路システム工学講座

〒739 東広島市鏡山一丁目 4-1

(電話) 0824-24-7662 (渡邊), -7661 (水野)

(ファクシミリ) 0824-22-7195

(電子メール) {mizuno,watanabe}@huis.hiroshima-u.ac.jp

概要：プリント基板レイアウト設計における基本的要求として、非平面接続（一層基板でのジャンパー線あるいは多層基板での異層間配線）を少なく維持しながら配線できることがある。本研究では、与えられた回路のグラフモデルから非平面接続を抽出する際に、接続要求を維持しながらグラフモデルを変形する操作に基づいてこれらを極小化する手法を提案し、その性能を実験的に評価する。

キーワード：プリント基板，レイアウト設計，グラフモデル，非平面接続，平面グラフ抽出

A Method of Reducing Nonplanar Connections in Layout Design of Printed Wiring Boards

Kenji Mizuno† and Toshimasa Watanabe††

†Graduate School of Engineering, Hiroshima University

††Department of Circuits and Systems, Faculty of Engineering, Hiroshima University

4-1, Kagamiyama 1-chome, Higashi-Hiroshima, 739 Japan

Phone : +81-824-24-7662 (Watanabe), -7661 (Mizuno)

Facsimile : +81-824-22-7195

E-mail : {mizuno,watanabe}@huis.hiroshima-u.ac.jp

Abstract: Routability with the number of nonplanar connections (jumpers in single-layered design or nets connecting different layers thorough vias in multi-layered one) kept small is one of basic requirements in layout design of printed wiring boards. The subject of the paper is to propose a method of reducing nonplanar connections in extracting a spanning subgraph from the graph model of a given circuit. It is based on transforming the graph model without changing connection requirements of the circuit. Experimental results are provided to show its capability.

Key words : Printed wiring boards, Layout design, Graph models, Nonplanar connections, Planar graph extraction

1 Introduction and Motivation

The paper proposes a method of reducing nonplanar connection (jumpers in single-layered design or nets connecting different layers thorough vias in multi-layered one) in layout design of printed wiring boards.

We have developing MULTI-PRIDE (MULTI-layered PRinted wiring DEsign system), a support system for designing multi-layered printed wiring boards [17, 18]. Here, a multi-layered printed wiring boards means one printed wiring board consisting of n layers for $n \geq 1$ (see Fig.1), where if $n \geq 3$ then the first and last layers are called *outside layers* and any other layer is called an *inside layer*. (In case of $n = 2$, the first layer is called an outside layer, and the last layer is called an inside one.) Elements are placed on one or both of the outside layers and routing is executed on outside layers and/or inside layers. Inside layers will be provided if any connection requirement remains unconnected after routing on outside layers: additional inside layers may be inserted one by one until all connection requirements are connected.

The designing flow of MULTI-PRIDE consists of (i) circuit bipartition, (ii) placement and routing on each outside layer (often followed by moving some elements to specified positions), (iii) routing on inside layers, (iv) modification of wiring and compaction. This paper is concerned with (i) and (ii), the latter of which is based on planar subgraph extraction and rectangular dualization.

Increase in the number of nonplanar connections causes deterioration in the quality of boards, a rise in manufacturing cost, and so on. Therefore, from practical points of view, sufficient routability with the number of nonplanar connections kept small is required as basic capability for printed wiring board design system.

There are some known algorithms that extract a spanning planar subgraph with maximal or almost maximal number of edges. See [2] for planarity testing and embedding, and [3, 6, 9, 10, 14] for planar subgraph extraction. Unfortunately they are unlikely to be useful in practical design process due to lack of capability of handling physical conditions specific to design of printed wiring boards. Therefore MULTI-PRIDE uses the algorithm PLAN-PWB or PLAN-MW. PLAN-PWB, proposed in [5], is one of vertex addition algorithms and uses PQR-trees [9] for finding a planar spanning subgraph of a given graph and for handling such physical conditions. PLAN-MW is based on the path addition algorithm and is proposed in [12, 13] for the similar purpose. It has capability of placing some elements to specified locations

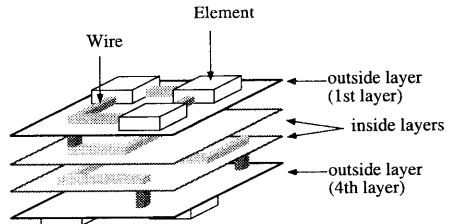


Figure 1: An example of one printed wiring board consisting of four layers (shown as if they were separated).

of a board. However those nonplanar edges, which are deleted from a given graph by these two algorithms, do not necessarily correspond to actual nonplanar connections.

The subject of the paper is to propose a method of reducing actual nonplanar connections in extracting a spanning planar subgraph from the graph model of a given circuit. It is based on transforming the graph model without changing connection requirements of the circuit. Experimental results provided show capability of the proposed method.

2 Preliminaries

2.1 Physical Conditions

Placement and routing on outside layers are considered under the following physical conditions 1-5.

1. There are three kinds of elements; *linear* elements (Fig.2), *up-sided* elements (Fig.3 (a)), each having a specified side which has to be faced to the board in actual mounting, and *free* elements (Fig.3 (b)) that have no such constraints. Usually a free element with just two terminals is represented as a linear one.
2. Each element is to be placed on one outside layer of a board, and each up-sided element must be placed as it should be.
3. Wires connecting different layers pass through vias. Wires on outside layers are called *outside wires* and those on inside layers are called *inside wires*.
4. No two wires on any one layer can cross each other.
5. Routing through *element-areas* (an area on the board to be occupied by an element) is prohibited. (We add this condition for simplicity of discussion, since this problem itself

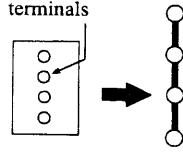


Figure 2: Representation of a linear element.

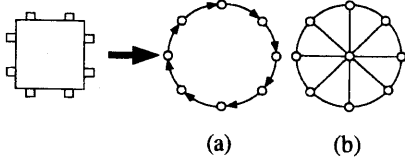


Figure 3: Representation of elements: (a) an up-sided element as a clockwise directed cycle; (b) a free element as a wheel graph.

contains an NP-complete problem [8], making the discussion too complicated. This restriction may be removed by incorporating postprocessing. Routing without this constraint is under investigation.)

2.2 Graph Models

We use a graph model called a *terminal-vertex graph* $G_T = (V_T, E_T)$. Suppose that a circuit is given by net lists. A maximal set of terminals requiring electrical connection among them is called a *net*, where distinct nets should have no electrical contact. If we are given a circuit (as in Fig.4) then we represent it as a graph G_T (as in Fig.5).

G_T is defined as follows. Represent each two-terminal element by an edge, and other elements by graphs as shown in Fig.2 for linear elements or in Fig.3 (a) for up-sided ones or (b) for free ones. These vertices are called *terminal vertices*, and we call these edges *non-removable edges*, which have to be included in any planar subgraph to be extracted. Each two-terminal nets is represented as a simple edge, and each of other multi-terminal nets is represented by a star-shaped steiner tree (defined by a new vertex, called the *net vertex*, and the edges connecting the net vertex and each of its terminal vertices). We call those edges for representing nets *net edges*.

In general, a multi-terminal net can be represented by a complete graph or a spanning tree. In case of representation by a complete graph, however, the number of nonplanar edges tends to increase in the subsequent planar subgraph extraction. If we adopt representation by a spanning

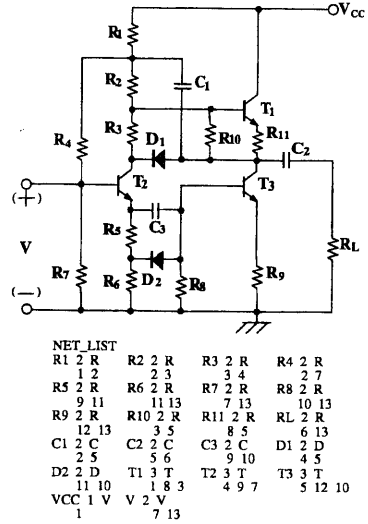


Figure 4: An example of a circuit and the set of net lists.

tree, the number of nonplanar edges greatly varies, depending upon the structure of the created tree for each net. Therefore we represent each multi-terminal net as a star-shaped steiner tree. The net vertex for each net represents routing area for the net, as will be shown by rectangular dualization in 2.3.

In this paper, without loss of generality, we assume that G_T is a connected graph. For a connected graph G and a cutpoint v of G , let V_1, \dots, V_i denote the vertex sets of connected components of $G - v$. The subgraph of G induced by each $V_i \cup \{v\}$ (or, for simplicity, its vertex set itself) is called a *v-block*.

2.3 Rectangular Duals

Given any circuit, we first partition it into two subcircuits by an existing bipartitioning algorithm [16], and construct G_T for each subcircuit. Then we extract a spanning planar subgraph from each graph model G_T . Some of nonplanar edges may correspond to actual nonplanar connections: others may be created because of representation of nets. Connection requirements represented by this planar subgraph are to be embedded without violating physical conditions. In order to handle placement simultaneously with routing, this planar subgraph is rectangular-dualized by adding some new vertices and edges so that the resulting graph may have a rectangular dual (such a planar graph is called a PTP graph: see [7, 18]).

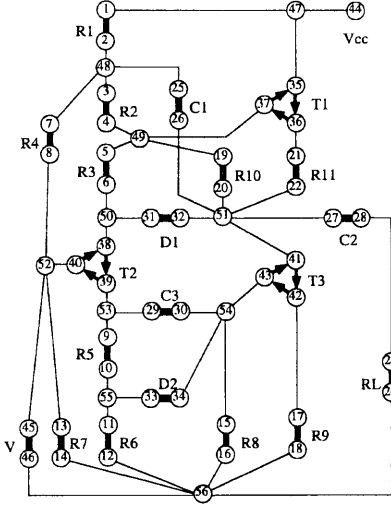


Figure 5: The terminal-vertex graph $G_T = (V_T, E_T)$ for the circuit in Fig. 4.

A rectangular dual $R(G)$ (see [1, 4, 7] for example) of a connected planar graph G correspond to a dissection of a whole rectangle into a set of subrectangles each of which represents a vertex of G , and two subrectangles share a side if and only if corresponding two vertices are adjacent in G . It has capability of controlling placement: making two subrectangles adjacent is done by adding an edge between corresponding vertices in G , and conversely placing two subrectangles apart can be realized by creating a path of appropriate length between corresponding two vertices of G , where addition of new vertices or edges may be required.

3 Reducing Nonplanar Connections

In this section, without loss of generality, we consider only single-layered design, since each outside layer design can be considered as single-layered one.

3.1 Basic Idea

Nonplanar edges found by planar subgraph extraction are net edges representing two-terminal or multi-terminal nets. In constructing the graph model of a given circuit, a multi-terminal net is represented by a star-shaped steiner tree where the net vertex is a steiner point. However, as mentioned in 2.2, multi-terminal nets do not have to be so represented. As long as connection requirements are incorporated, any representation of a multi-terminal net will do.

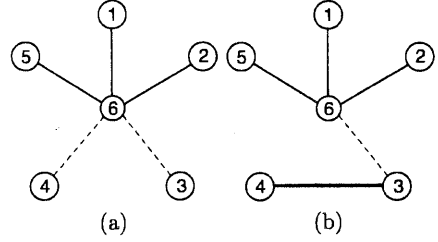


Figure 6: Representation of a multi-terminal net (vertices 1 through 5 are the terminal vertices and the vertex 6 is the net vertex): (a) a star-shaped steiner tree; the edges (3,6) and (4,6) are nonplanar edges corresponding to virtual nonplanar connections; (b) after transformation, (3,6) is a nonplanar edge corresponding to an actual nonplanar connection.

In the example of Fig. 6 (a), we assume that edges (3,6) and (4,6) are removed by planar subgraph extraction as nonplanar connections. If we can transform (a) to (b), then one nonplanar connection will be excluded without changing connection requirement, leaving (3,6) as the only nonplanar connection. This is the basic idea of the method to be proposed. In other words, if some nonplanar edges are found, then we consider them as *virtual nonplanar connections*. *Actual nonplanar connections* (actually processed as nonplanar connections) will be fixed by transforming representation of multi-terminal nets with connection requirements kept unchanged. This is how we reduce the number of nonplanar connections.

We try to detect possibility of such transformation by processing faces in a plane graph one by one. A *face* of a plane graph is a maximal region of the plane such that, for every two points x and y in it, there is a continuous line from x to y which does not contain any point in the outside. If there is any face whose contour contains at least two terminal vertices belonging to the same net then we can add planar connections among them. Therefore we can increase planar connections, leading to reduction of nonplanar connections.

We call a net *npc-reducible* (with respect to the current plane graph) if and only if there is any face whose contour contains at least two terminal vertices of the net. A face f is called *npc-decreasing* if and only if there is a net N with $V_f(N) \neq \phi$.

For example, suppose that the multi-terminal net of Fig. 6 (a) is located as in Fig. 7 (a). In Fig. 7 (a), we can exclude one nonplanar connection by means of the transformation from (a) to (b) of Fig. 6: by adding a planar edge between terminal

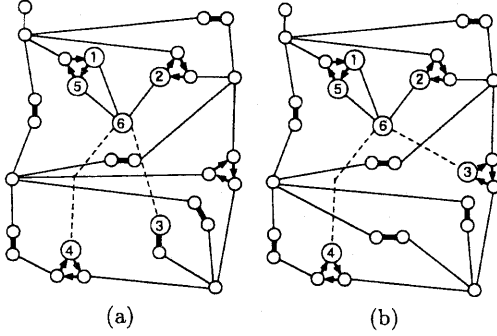


Figure 7: Detection of npc-reducibility of the multi-terminal net of Fig.6 (a): (a) npc-reducible; (b) npc-irreducible case.

vertices 3 and 4 and by deleting the edge (4,6). On the other hand, since terminal vertices 3 and 4 exist in the contours of the different faces in Fig.7 (b), we cannot add a planar edge between them and, therefore, the number of nonplanar connections cannot be reduced.

In the following section, we explain details of procedure MNC for reducing nonplanar connections by finding npc-reducible nets and npc-decreasing faces.

3.2 Procedure MNC

We fix any terminal-vertex graph $G_T = (V_T, E_T)$. Let V_n or V_t denote the set of net vertices or that of terminal vertices, respectively, where $V_T = V_n \cup V_t$. Let E_{np} denote any set of nonplanar edges found by planar subgraph extraction from G_T . Let E_n denote the set of net edges, and let $G_{planar} = (V_n \cup V_t, E_T - E_{np})$ denote the resulting spanning planar subgraph of G_T . Let us fix any plane embedding of G_{planar} . For any face f of G_{planar} and any net $N \subseteq V_t$, let $V_f(N)$ denote the set of terminal vertices contained in the contour $\alpha(f)$ of f , and let

$$V_f = \bigcup_N V_f(N).$$

A face f is *adjacent* to another one f' if and only if $\alpha(f)$ shares at least one vertex with $\alpha(f')$. For any face f , $\alpha(f)$ consists of one cycle $\tau(f)$, called the *main cycle* of $\alpha(f)$, and several w -blocks for some cutpoints w contained in $\alpha(f)$. Here w -blocks are called *attachments* of $\tau(f)$ at w . Let f' be any face adjacent to f , and suppose that there is any attachment B_w of $\tau(f')$ at w contained in $\tau(f')$ and $\tau(f)$. The subcontour of $\alpha(f')$ enclosing B_w (as if it were the contour of the external face of B_w) is denoted by $\beta_w(f'; f)$ and is called an *attaching subcontour* (of $\alpha(f')$ at w).

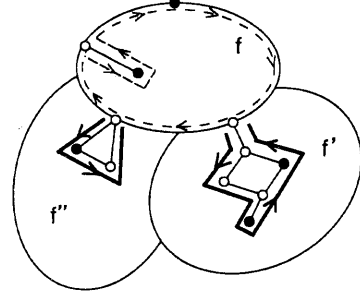


Figure 8: Illustration of step3. (Each black vertex shows that it satisfies the condition (a) or (b) in step2.)

First we find V_f for each face f except those whose contours are directed cycles representing up-sided elements of G_{planar} . This is done by setting $\theta \leftarrow \alpha(f)$ initially and then by executing the following steps 1-4 at each terminal vertex v of θ . If $\theta = \alpha(f)$ then we proceed θ clockwise.

step1. Set $V_f \leftarrow \phi$.

step2. If v is a cutpoint then goto step3. Else if v satisfies one of the following two conditions (a) and (b), then set $V_f \leftarrow V_f \cup \{v\}$.

- (a) a nonplanar edge is incident upon v .
- (b) v is a terminal vertex upon which exactly two non-removable edges of a linear element having at least three terminals are incident.

step3. For any face $f' (\neq f)$ adjacent to f , execute step2 by setting $\theta \leftarrow \beta_w(f'; f)$ for each cutpoint w contained in $\alpha(f')$ and θ , where if $\theta \neq \alpha(f)$ then we proceed θ counterclockwise. (This may create recursive calls to the step as shown by bold lines of Fig.8. Reversing the direction is required in order to properly handle directed cycles representing up-sided elements.)

/* If $V_f = \phi$ then f is not npc-decreasing. */

step4. If $V_f \neq \phi$ then delete any terminal vertex satisfying one of the following two conditions from V_f :

- (i) v is a terminal vertex with $V_f(N) = \{v\}$ for some net N ;
- (ii) v is a terminal vertex of a net N such that $V_f(N)$ includes only vertices satisfying the condition (b) of step2.

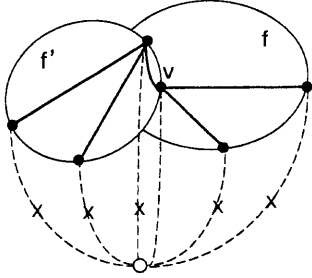


Figure 9: Fixing actual nonplanar connections. (Broken lines marked with X represent edges excluded from E_f , and bold lines represent edges in E_{add} . The open vertex represents a net vertex.)

/* Consequently if $V_f = \phi$ then f is not npc-decreasing. */

Any vertex v of V_f is included in $\alpha(f)$ or in an attachment of $\tau(f')$ at some cutpoint w' contained in $\tau(f')$ and $\tau(f)$, where f' is adjacent to f (see Fig.8).

Define the cost $c_f(v)$ of any vertex v of V_f by the following (1)-(3).

- (1) If v_1, \dots, v_m are terminal vertices added to V_f during the search in any one attachment of $\tau(f)$ at a cutpoint w contained in $\tau(f)$ and if v is equal to some v_j then we set $c_f(v) = m$.
- (2) If v is contained in $\tau(f)$ then we set $c_f(v) = 0$.
- (3) If v_1, \dots, v_n are terminal vertices added to V_f during the search in any one attachment of $\tau(f')$ at a cutpoint w' in $\tau(f')$ and $\tau(f)$, and if v is equal to some v_j then we set $c_f(v) = n$.

Secondly, if $V_f \neq \phi$ then we number vertices of V_f in the order of their addition into V_f during the search along $\alpha(f)$. Now we will find one or more sets $E_f^{(i)}$ of edges satisfying the following three conditions (i)-(iii).

- (i) Each edge in $E_f^{(i)}$ is a planar edge between terminal vertices of $V_f(N) \subseteq V_f$ for some net N .
- (ii) $G = (V_n \cup V_t, \{E_n - E_{np}\} \cup E_f^{(i)})$ is a simple graph, where E_n is the set of net edges of G_T .
- (iii) No cycles exist in G .

Let $V_{add}^{(i)}(f)$ denote the set of endvertices of edges in $E_f^{(i)}$ and let

$$cost(f; E_f^{(i)}) = \frac{\sum_{v \in V_{add}^{(i)}(f)} c_f(v)}{|E_f^{(i)}|}.$$

Let E_f be any edge set chosen as follows:

$cost(f; E_f)$ is minimum among those $cost(f; E_f^{(i)})$ with $|E_f|$ being maximum among $|E_f^{(i)}|$. We can find such E_f by extending the algorithm which is proposed in [15] for finding a maximum independent set of a circle graph, based on the dynamic programming.

Thirdly we construct an edge set E_{add} based on $cost(f; E_f)$. Edges of E_{add} are to be added to G_{planar} without violating planarity. We set $E_{add} = \phi$, and find a face, having the smallest cost among those npc-decreasing faces not yet processed, as the target face f for which the following operation is executed:

$$E_{add} \leftarrow E_{add} \cup E_f,$$

$$G_{planar} \leftarrow G_{planar} + E_f,$$

and update faces and cutpoints of G_{planar} .

These operations are repeated until no npc-decreasing faces exist.

After executing the above operations, $G = (V_n \cup V_t, \{E_n - E_{np}\} \cup E_{add})$ is necessarily a forest $F = \{T_1, T_2, \dots, T_m\}$. For each tree T_i , we execute the following operation starting from an arbitrary vertex of T_i .

- If T_i contains any net vertex then delete from E_{np} all edges incident upon vertices in T_i . Otherwise select an arbitrary vertex v in T_i and delete from E_{np} every edge incident upon any vertex except v of T_i (Fig.9).

We can reduce nonplanar connections by the number of edges in E_{add} finally obtained. Edges left in E_{np} are to be actual nonplanar connections (Fig.9).

3.3 Application to Circuit Bipartitioning

A circuit-bipartitioning algorithm that tries to minimize the number of vias of multi-layered printed wiring board design is proposed in [16]. MULTI-PRIDE uses this algorithm. The main point of this algorithm is that a bipartition having small number of nonplanar connections is obtained. This is because reducing the number of nonplanar connections, even if it may result in increase in the number of cutnets, has possibility of leading to a layout with smaller number of vias. This algorithm

first obtains an initial bipartition of a given circuit and then improves this partition by moving vertices from one block to the other and by extracting a spanning planar subgraph. Improving an initial bipartition uses a measure in order to prevent increase in the number of nonplanar edges. The idea behind this is that reducing the number of nonplanar edges leads to decreasing the number of nonplanar connections and, therefore, the number of vias.

This algorithm considers every nonplanar edge to be an actual nonplanar connection. However a nonplanar edge found by planar subgraph extraction does not necessarily correspond to an actual nonplanar connection. In other words, "nonplanar connections" in this algorithm should be considered as virtual nonplanar connections in procedure MNC. Since this procedure reduces nonplanar connections, we may obtain a desired bipartition having smaller number of vias, by executing procedure MNC after planar subgraph extraction.

4 Experimental Results

The proposed method have been implemented on a personal computer GATEWAY2000 (CPU: Pentium/120MHz, OS: FreeBSD 2.1) with the C programming code. We use actual circuits and randomly generated ones as input data. Actual circuits data (given by net lists) are taken from audio circuits.

For each actual circuit, Table 1 shows the number of virtual nonplanar connections by PLAN-PWB and actual ones by procedure MNC in the columns "V_NC" and "A_NC", respectively, where the numbers of vertices and edges of the graph models are given in the column "#node" and "#edge", respectively. The column "CPU(s)" shows the computation time (in second) spent by procedure MNC.

For each randomly generated circuit, Table 2 shows the average number of virtual nonplanar connections by PLAN-PWB and actual ones by procedure MNC in the column "V_NC" and "A_NC", respectively, over 30 input data (15 input data in the case1 and 15 input data in the case2), where the graph models have the number of vertices in the intervals given in the column "#node". The column "ratio" shows the ratio A_NC / V_NC . The column "case1" ("case2", respectively) shows that the number of terminals in each net is no more than 10 (no more than 50).

For each input data, Table 3 shows the number of cutnets and of actual nonplanar connections in the column "CN" and "NC", respectively, where input data have the number of terminals and of

Table 1: Experimental results for actual circuits.

	#node	#edge	V_NC	A_NC	CPU(s)
data1	113	83	9	9	0.04
data2	141	104	6	6	0.05
data3	229	176	9	9	0.11
data4	242	188	21	20	0.23
data5	233	173	21	20	0.27
data6	285	216	24	22	0.39
data7	146	108	17	14	0.14
data8	142	114	14	14	0.07
data9	357	285	28	27	0.38
data10	276	212	41	23	0.96

Table 2: Experimental results for randomly generated circuits.

#node	case1			case2		
	V_NC	A_NC	ratio	V_NC	A_NC	ratio
1-100	12.4	11.8	0.95	11.4	10.6	0.93
101-150	21.8	20.4	0.94	19.6	18.2	0.93
151-200	27.6	26.2	0.95	33.8	30.2	0.89
201-250	40.8	37.8	0.93	45.0	41.4	0.92
251-300	57.2	51.8	0.91	52.4	44.8	0.85
301-350	88.0	81.2	0.92	83.2	71.6	0.86
351-400	99.2	89.6	0.90	108.2	87.8	0.81

nets given in the column "#terminal" and "#net", respectively. The column "UW" or "UW-MNC" shows the results by the UW method or those by the UW method followed by procedure MNC.

For data10 of Table 1, the large number of nonplanar connections excluded. The circuit of data10 has two nets such that 9 out of 12 nonplanar edges in the first net and 4 out of 8 ones in the second net are excluded by the proposed method. Table 2 shows that as the number of vertices or that of terminals of one net becomes larger then greater number of nonplanar connections are excluded. Therefore it seems that procedure MNC is effective if many nonplanar edges are included in one net. Table 3 shows that better bipartitions can be obtained if input data have many nonplanar connections. These experimental results show that the proposed method has high capability to reduce nonplanar connections. Also observed is that its application to circuit bipartitioning can improve quality of bipartition.

5 Concluding Remarks

This paper has proposed a method of reducing actual nonplanar connections in extracting a spanning planar subgraph from the graph model of a given circuit, based on transforming the graph

Table 3: Experimental results on circuit bipartitioning.

	#terminal	#net	UW		UW-MNC	
			CN	NC	CN	NC
actual1	240	64	21	15	21	14
actual2	117	28	19	6	19	6
random1	140	10	8	15	8	14
random2	170	27	22	25	23	22
random3	220	31	19	55	17	50
random4	205	54	25	38	24	35
random5	315	59	38	55	35	53
random6	275	43	19	29	19	29

model without changing connection requirements of the circuit. Experimental results provided show capability of the proposed method.

Some problems left for future research are as follows:

- improving the proposed method so as to reduce more nonplanar connections;
- providing a better measure for the searching order of faces in a plane graph.

Acknowledgments

The research of T.Watanabe is partly supported by the Grant in Aid for Scientific Research of the Ministry of Education, Science and Culture of Japan: under (C) 08680371 and (A) 07308028.

References

- [1] Bhasker, J. and Sahni, S., "A linear algorithm to find a rectangular dual of a planar triangulated graph", Technical Report TR 85-26, Computer Science Dept., University of Minnesota, Minneapolis, 1985. Also appeared in Proc.23th Design Automation Conference, pp. 108-114, 1986.
- [2] Booth, K. S. and Lueker, G. S., "Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms", J. Comput. & Syst. Sci., 13, pp. 335-379, 1976.
- [3] Chiba, N., Nishizeki, T., Abe, S. and Ozawa, T., "Embedding Planar Graphs Using PQ-Tree Algorithms", Trans. IEICE Vol.J67-A No.2, pp. 87-94, 1984. (in Japanese)
- [4] He, X., "On finding the rectangular duals of planar triangulated graphs", SIAM J. Comput., 22(6), pp. 1218-1226, 1993.
- [5] Iwamoto, K., Watanabe, T., Araki, T. and Onaga, K., "Finding Jumpers in Printed Wiring Board Design for Analog Circuits", Proc.1991 IEEE Int. symposium on Circuits and Systems, pp. 2854-2857, 1991.
- [6] Jayakumar, R., Thulasiraman, K., and Swamy, M.N.S., "O(n²) Algorithms for Graph Planarization", IEEE Trans. Computer-Aided Design, Vol.8, No.3, 1989.
- [7] Kzominski, K. A. and Edwin, K., "Rectangular Dualization and Rectangular Dissections", IEEE Transactions on Circuits and Systems, Vol.15, pp. 1401-1416, 1988.
- [8] Masuda, S., Kashiwabara, T. and Fujisawa, T., "On Single Layer Routing under the Condition that Routing is Allowed in the Area Occupied by the Module", Tech. Rep. of IEICE of Japan, CAS82-114, pp. 35-40, 1982. (in Japanese)
- [9] Masuda, S., Kashiwabara, T. and Fujisawa, T., "A Wiring Problem on Single Layer Printed Circuit Board without Mounting Modules Upside-Down", IEICE Trans. Vol.J66-A NO.3, pp. 235-242, 1983. (in Japanese)
- [10] Matsumoto, A., Yamaguchi, K., Kashiwabara, T., Masuda, S., and Taki, M., "A Planarization Algorithm in the Layout Design of Single Layer Printed Circuit Board", Tech. Rep. of IEICE of Japan, COMP91-21, pp. 79-88, 1991. (in Japanese)
- [11] Mizuguchi, Y. and Watanabe, T., "Application of the Path-Addition Planarity Testing Algorithm to Layout Design of Printed Wiring Boards", Tech. Rep. of IEICE of Japan COMP94-87, pp. 103-112, 1995.
- [12] Mizuguchi, Y., "A Study on Placing Elements at Specified Positions in Layout Design of Printed Wiring Boards", Master's thesis of Information Engineering Major, Graduate School of Engineering, Hiroshima Univ., 1995.
- [13] Mizuno, K. and Watanabe, T., "A Method of Reducing Jumpers in Layout Design of Printed Wiring Boards", Proc. of the 46th Joint Convention at the Chuugoku Branch of Electrical 5-Societies of IEICE of Japan, pp. 394-395, 1996. (in Japanese)
- [14] Ozawa, T. and Takahashi, H., "A graph-planarization algorithm and its application to random graphs", in Graph Theory and Algorithms, Lecture Notes in Computer Science, Vol.108, Springer-Verlag, pp. 95-107, 1981.
- [15] Supowit, K. J., "Finding a Maximum Planar Subset of a Set of Nets in a Channel", IEEE Transactions on Computer-Aided Design, vol.CAD-6, No.1, pp. 93-94, 1987.
- [16] Une, Y., Mizuguchi, Y. and Watanabe, T., "A Circuit-Bipartitioning for Multi-Layered Printed Wiring Board Design", IPSJ, SIG Notes, IPS of Japan 94-DA-70, pp. 47-54, 1994. (in Japanese)
- [17] Yasui, T., "A Support System MULTI-PRIDE for Designing Multi-Layered Printed Wiring Boards", Master's thesis of Information Engineering Major, Graduate School of Engineering, Hiroshima Univ., 1993.
- [18] Yasui, T., Tōyama, N., Une, Y., Watanabe, T. and Onaga, K., "MULTI-PRIDE: A Support System for Designing Multi-Layered Printed Wiring Board Layouts of Analog Circuits", Proc. DA Symposium'92, IPS of Japan, pp. 137-140, 1992. (in Japanese)