AN ALGORITHM FOR TWO-LAYER CHANNEL ROUTING WITH CYCLIC CONSTRAINTS

by

M. L. Liu

Memorandum No. UCB/ERL M81/82

17 November 1981

AN ALGORITHM FOR TWO-LAYER CHANNEL ROUTING WITH CYCLIC CONSTRAINTS

by

M. L. Liu

Memorandum No. UCB/ERL M81/82

17 November 1981

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

AN ALGORITHM FOR TWO-LAYER CHANNEL ROUTING WITH CYCLIC CONSTRAINTS*

by

M. L. Liu**

ABSTRACT

Two-layer channel routing is one of the key steps in the automatic layout design of VLSI chips. There are several efficient algorithms developed for channel routing. But usual channel routing algorithms do not work when the vertical constraint graph is cyclic.

A extension of two-layer channel routing are presented based on a merging algorithm on the vertical constraint graph. It can deal with the cyclic problems. Five subalgorithms have been developed. The program goes through the subalgorithms until a solution is obtained. The last subalgorithm calls for the use of extra vertical track beyond the end of the channel. With that, a solution is guaranteed.

The algorithm has been programmed in PASCAL language and implemented on a VAX 11-780 computer. Experimental results are quite encouraging.

---

*Research supported by the National Science Foundation Grant ENG-78-24425.

**On leave from Tianjin University, Tianjin, China.

AN ALGORITHM FOR TWO-LAYER CHANNEL ROUTING WITH CYCLIC CONSTRAINTS

M. L. Liu

1. Introduction

Two-layer channel routing is employed frequently in the LSI chips layout [1] [2] and [3]. Two rows of cells are placed on two sides of a channel. Along the channel, every terminal of cells has a certain number, and terminals with the same number must be connected by a net, while terminals with number 0 are left unconnected. Usually we assume the horizontal segments of nets on one layer and the vertical segments of nets on the other. Fig. 1-1 shows an example of a two-layer channel routing problem. The real lines are on one layer and the dotted lines are on the other. The black square spots represent via holes connecting the horizontal and vertical segments of a net.

Because the vertical segment of one net can not overlap that of another on the same track, a constraint has been introduced on horizontal segments of the nets. For example, in Fig. 1-1, if we pay attention to the rightmost column, the horizontal segment of net 10 must be placed above that of net 9. This relation can be represented by a directed graph, so-called vertical constraint graph -- V.C.G., where each node corresponds to a net and a directed edge from node A to node B means that net A must be placed above net B. Fig. 1-2 shows the V.C.G. of the example in Fig. 1-1.

In the algorithm described in this paper, the dogleg is allowed. This means that the horizontal segment of a net can be divided at its terminal positions. In the case with the dogleg, the subnets are introduced whose horizontal segments are parts of a net between two consecutive terminals of a net. Henceforth a subnet will also be called a 'net'. For the dogleg problems, there are more nodes in the V.C.G. because of the subnets. However, the degree of the nodes (the number of edges connecting to every node) is less than or equal to that for non-dogleg problems. The highest degree of any node in the V.C.G. is four for the dogleg problems. Fig. 1-3 shows an example, its V.C.G.'s (without dogleg or with dogleg) are shown in Fig. 1-4(a) and Fig. 1-4(b).

Sometimes, there exists a cycle in the V.C.G., i.e. there is a directed path starting from a node, going though others and ending at the node itself. For example, in Fig. 1-5(a), there is a cyclic directed path A-B-A. It means that net A should be above net B and net B should be above net A. Obviously, this requirement can not be realized. Fig. 1-5(b) and Fig. 1-5(c) show other cyclic examples.

In some problem, the cycle can be resolved by using doglegs. For example, Fig. 1-6(a) shows a non-dogleg problem; its V.C.G. is shown in Fig. 1-6(b). The same problem but with the dogleg and its V.C.G. are shown in Fig. 1-6(c) and Fig. 1-6(d). In this example, when the dogleg is used, the cycle is resolved. But the dogleg can not always resolve a cycle. As in Fig. 1-5, none of the cycles can be resolved by using doglegs.

In this paper, an effective algorithm will be proposed for the channel routing problems in which there are cycles that can not be resolved by using doglegs.

2. Some definitions

Before the description of the algorithm, some definitions are needed:

A. The involved nets (subnets) of a cycle: If there is a cyclic directed path through some nodes in the V.C.G., these nodes are called the involved nodes of that cycle and the corresponding nets (subnets) of these nodes are called the involved nets (subnets) of that cycle. For example, in Fig. 2-1 the involved nets of the cycle are nets A, B and C.

B. The exposed node: In the V.C.G., some nodes have the directed edges all with the same direction. That is, all its edges point out from the node or all its edges point in to the node. This kind of node is called an exposed node. As in Fig. 2-2, the dark nodes are exposed nodes. In the situation with the dogleg, every subnet has two and only two terminals. If the two terminals of a subnet connect to the same side of the channel (both to the upper side of the channel or both to the lower side of the channel), the corresponding node of that subnet in the V.C.G. will be an exposed node. It is obvious that the involved nodes of a cycle can not be exposed nodes; in other words, an exposed node can not be involved in a cycle.

C. The connected subnets of a subnet: If some subnets have the same terminal number with a certain subnet, they are called the connected subnets of the subnet. As in Fig. 2-1, subnets D, E and F are the connected subnets of subnet A, all of which have the same terminal number '1'.

D. The range of a cycle: The range between the leftmost starting column of the involved subnets of a cycle and the rightmost ending column of the involved subnets of this cycle is called the range of the cycle.

E. The extension of the range of a cycle: The range between the leftmost starting column of the connected subnets of the involved subnets of the cycle and the rightmost ending column of the connected subnets of the involved subnets of the cycle is called the span of the cycle. The span of a cycle minus the range of this cycle is called the extension of the range of the cycle. This is illustrated in Fig. 2-1.

F. The semi-empty column: A column with one unconnected terminal (either upper or lower) is called the semi-empty column. Generally a column with both upper and lower terminals unconnected is also considered as the same as a semi-empty column unless specially mentioned.

3. The main idea of the algorithm

When the algorithm was proposed, there were some main considerations:

A. The merging algorithm as in reference [2] is still employed here. Since there are some cycles in the V.C.G., before the merging procedure some of the involved nets will be modified for breaking the cycles.

B. For the sake of saving the cpu time and the memory, the cycles will be treated one by one no matter whether there are some relationships between any two cycles or not.

C. For every cycle, only one involved net will be modified.

D. As reference [4] has shown, the lower bounds of the double-layer channel routing problems are both dmax and vmax (the max density and the max level of the V.C.G.). The algorithm will attempt to minimize the increase of either dmax or vmax.

E. When a net is modified (breaking, extending, etc.), its starting or ending column changes. Hence its relative vertical constraint will change. In the V.C.G. some of its related edges will disappear and some new edges will be introduced. To avoid a sizeable increase in the vmax, when the modification is done, the additional dogleg must appear only on a semi-empty column or an extra column. An opposite situation is shown in Fig. 3-1 where a new dogleg is introduced on a non-empty column and eight new edges appear.

F. Five methods are proposed for resolving the cycles. For every cycle, the methods will be tried one by one in sequence. If one of the five methods succeeds, the rest of the methods will not be tried for that cycle.

4. Method 1 -- Break an involved net in the range of the cycle

This is the best method among those presented in this paper for solving a cyclic constraint problem. For instance, Fig. 4-1(a) can be solved by this method as in Fig. 4-1(b). One of the involved nets of the cycle is broken into two at a semi-empty column within the range of the cycle (and the column is between the starting column of the net and the ending column of the net). In this approach, the density of only one column changes -- the density of that column for breaking the net increases by one. Hence, generally the max density does not increase.

For this approach, the procedure contains two steps: choosing a suitable semi-empty column, and choosing an involved net which is going to be broken. A suitable semi-empty column is chosen according to the following conditions:

A. It should be within the range of the cycle.

B. It is supposed to have at least one empty terminal (either upper or lower).

For the sake of simplicity, when the additional conditions are described, it will be assumed that the upper terminal of the column is empty. If the lower terminal is non-empty, at most two nets (subnets) connect to it. Fig. 4-1(a) shows an example, net P and net Q connect to the lower terminal of the semi-empty column. It should be mentioned that, henceforth, the names 'net P' and 'net Q' will be used for the nets which connect to the terminal of a semi-empty column.

It can be seen that, as in Fig. 4-2(b), when net X is broken into X' and X", some new directed edges will be added from X' and X" to P and Q in the V.C.G., and there will still be a directed path from X" to X' through all the involved nodes of the cycle. If originally there was a directed path from P or Q to any of the involved nodes of the cycle, there will be a new cycle formed. Therefore another condition is needed:

C. There must be no directed path from either node P or node Q to any of the involved nodes of the cycle in the V.C.G.

After net X was broken, the max length of the directed path from X" through P or Q to the bottom of the V.C.G. (as in Fig. 4-3) is determined as follows:

$$L=L1+L2+1,$$

Where:

L1 is the length of the cycle which is equal to the number of the involved nets of the cycle;

L2 is the length of the longest directed path from node P or Q to the bottom of the V.C.G.

To avoid a sizeable increase in the vmax, another condition is:

D. If there are more columns than one satisfying the above three conditions, the one will be selected whose length of the longest path from node P or node Q to the bottom of the V.C.G. is the smallest among those columns .

The second step is determination of the net which will be broken. When the suitable semi-empty column has been chosen, at least two nets crossing this column are available, the procedure selects any of them arbitrarily.

5. Method 2 ------ Extend an involved subnet to another terminal

Generally, the horizontal segment of a subnet is only between two consecutive terminals which have the same terminal number, as in Fig. 5-1(a). This makes the horizontal segment of every subnet as short as possible and thus guarantees less density for every column.

In method 2, one involved subnet of a cycle will change its starting or ending column and extend leftward or rightward to another terminal that has the same number. In this approach, the density of some columns will increase by one, but the number of subnets will not increase and the number of doglegs required will be the smallest among the five methods presented in this paper. These are its advantages. From definition B in section 2, it should be remembered that an exposed node can not be involved in a cycle. Method 2 requires that when subnet X (as in Fig. 5-1) is extended, it should connect to a terminal which goes to the same side to which the remaining terminal of subnet X goes. Then node X will become an exposed node; the cycle will be resolved and no new cycle can be formed.

If there are more involved subnets than one which can be extended to some suitable terminals, the one that requires the least extension will be selected.

6. Method 3 --- -- Use a semi-empty column within the extension of the range of the cycle

If the two previously described methods can not be used, a suitable semi-empty column within the extension of the range of the cycle may be employed. For example, in Fig. 6-1(a), if net X is an involved net of the cycle, it can be extended rightward or leftward to a semi-empty column in the extension of the range of the cycle and connect to a net C which is a connected net of net X, as in Fig. 6-1(b) or (c).

In this approach, the density of some columns increases by one.

To determine the suitable semi-empty column, there are four conditions. These conditions are the same as in method 1 except condition A: the column should be within the extension of the range of the cycle and should be as near the range of the cycle as possible.

After the suitable semi-empty column has been determined, the net to be extended (as in Fig. 6-1 net X) must meet the following conditions:

1. One of its connected subnets must cross the column selected.

2. There must be no directed path from net P or net Q to net C. Otherwise, because when net X connects to net C, there will be some additional vertical constraints between net C and net P, Q, and a new cycle will be formed.

3. The distance from the original starting column of net X for extending leftward or the original ending column of net X for extending rightward, to the suitable semi-empty column should be as short as possible.

4. If a net is extended to the column selected and its corresponding node becomes an exposed node, then that net will have priority for selection.

Although method 3 seems similar to method 2, they have some different properties:

a. In method 3, the net X is extended to a semi-empty column; but in method 2, the net X is extended to a terminal column of its connected subnet.

b. In method 3, the node X does not necessarily become exposed; but in method 2, the node X must become an exposed node.

c. In method 3, the extended net connects to the horizontal segment of a net via a dogleg; but in method 2, the extended net connects to the vertical segment of a net.

d. In method 3, both the number of doglegs and the number of the via holes are one more than those in method 2.

7. <u>Method 4</u> -- --<u>Use a semi-empty column outside the span of the cycle</u>

If a suitable semi-empty column can not be found even in the extension of the range of the cycle, a semi-empty column outside the span of the cycle may be found. In this approach, the density of some columns increases by one or two. The max density may also increase. Hence, the semi-empty column selected should be as near as possible to one or the other end of the span of the cycle.

For choosing the suitable semi-empty column, the conditions are similar to those of method 1 and method 3, as follows:

A. It should have at least one empty terminal, assume the upper terminal is empty in the following descriptions.

B. There must be no directed path from either node P or node Q to any of the involved nets of the cycle.

C. The length of the longest path from node P or node Q to the bottom of the V.C.G. should be as small as possible.

D. If there are more columns than one satisfying the above three conditions, the column nearest to one or the other end of the span of the cycle will be selected.

After the suitable semi-empty column has been chosen, the next step is selecting the net which will be extended.

Because the density of the columns between the semi-empty column selected and the farthest connected net of net X (as in Fig. 7-1 net E) will increase by two. In method 4, the net will be selected whose farthest connected net (that means net E) is as near as possible to the column selected. And, if a net is extended to the determined column and its corresponding node becomes an exposed node, that net will have the priority for selection.

It can be seen that, when the additional net Y is connected to net E, there will be some new vertical constraints between node Y and nodes U and V (as in Fig. 7-2). Therefore, another condition appear: there must be no directed path from node P or node Q to node U or node V.

8. <u>Method 5</u> ----- <u>Use an extra column</u>

If a cycle can not be solved by using any of the methods described in the four preceding section, an extra column will be added to the left or right end of the original range of the channel for breaking one of the nets involved in the cycle (Fig. 8-1) . This would be used instead of the suitable semi-empty column in method 4.

In this approach, not only the density of some columns will increase by one or two, but also the number of columns as well as the length of the channel will increase. However, because it does not need any semi-empty columns within the channel and there are no new restrictions in the V.C.G., method 5 can resolve any kinds of cycles as long as the problem allows some extra columns to be added.

The only things to be specifically mentioned here are:

A. The extra column is added to the left or right end of the channel according to which end of the channel is nearer the span of the cycle. And the net selected to be extended to the extra column is the one whose connected net is nearest to the column. This approach will guarantee the least extensions of nets.

B. Since the dogleg on the extra column is usually not short -- in some practical problems it is often more than half of the width of the channel - every extra column is employed for only one cycle. If there are more cycles, more extra columns will be added for them, respectively. Thus, this approach can avoid some new vertical constraints caused by the extra columns, and can avoid increasing the height of the V.C.G.

9. Modification of the original data

When the method for breaking a cycle is determined, the original data should be modified.

For all methods except method 2, one of the nets involved in the cycle will become two. There will be a new net (subnet) and the number of nets (subnets) will increase by one. Because some nets will change their starting or ending column, the zone representation will be modified. Furthermore, the vertical constraint graph will also be modified. Some directed edges will be deleted; some new directed edges will be added; and most often a new node will appear. After that the cycle will be broken.

Besides the above modifications the semi-empty column or extra column employed will now be marked as occupied. Hence it can not be employed again.

10. The combined algorithm

The complete algorithm includes three parts. The first part is for data input and pre-processing. The second part is for finding whether there are some cycles or not, and how to break the cycles. The third part is for routing and plotting, that is the same as in usual merging algorithm [2]. The detail of them is described as follows:

Part 1. Data input and pre-processing:

A. Input the data.

B. Count the number of nets (subnets) and determine the starting column and the ending column of every net.

C. Determine the zone representation of every net.

D. Construct the vertical constraint graph.

Part 2. Finding and breaking of the cycles:

Step A. Find whether there is a cycle or not. If yes, go to next step; otherwise go to part 3.

Step B. Try to break the cycle by using method 1 (section 4). If it succeeds, return to step A to see if there is another cycle; otherwise go to next step.

Step C. Try to break the cycle by using method 2 (section 5). If it succeeds, return to step A to see if there is another cycle; otherwise go to next step.

Step D. Try to break the cycle by using method 3 (section 6). If it succeeds, return to step A to see if there is another cycle; otherwise go to next step.

Step E. Try to break the cycle by using method 4 (section 7). If it succeeds, return to step A to see if there is another cycle; otherwise go to next step.

Step F. Try to break the cycle by using method 5 (section 8). It can always succeed. Return to step A to see if there is another cycle.

Part 3. Routing and plotting of the channel:

The merging algorithm is the same as in [2]. The detail is omitted here.

The flowchart diagram of the whole program is shown in Fig. 10-1.

11. Conclusion

In this paper, an algorithm is proposed for the two-layer channel routing. It can deal with any input data no matter whether there are some cycles in the V.C.G. or not.

In the algorithm, there are five different methods of breaking a cycle. For a cycle in the V.C.G., the five methods will be tried one by one. If one of the first four methods succeeds, the problem can be routed within the length of the channel, otherwise an extra column will be added.

The algorithm has been programmed in PASCAL and run on a VAX 11/780 computer. Experimental results are quite good. The cpu time for dealing with a cycle is trivial compared with the cpu time for routing. Most of the practical problems tried can be routed without the extra column.

Fig. 11-1 to Fig. 11-11 show some computational results of the program.

It should be pointed out that this algorithm can be employed not only by the general two-layer channel routing problems (with either merging algorithm or matching algorithm introduced in reference [2]), but also by the channel routing problems with irregular boundaries (as in reference [3]) and even by the three-layer channel routing problems (as in reference [4]). The programs for irregular boundaries or for three-layer routing are going to be written.

REFERENCES

[1] D. N. Deutsch, C. Persky and D. G. Schwiekert: "LTX-A System for the Directed Automatic Design of LSI Circuits" . Proc. 13th Design Auto. Conf. 1976, pp. 399-407.

[2] T. Yoshimura and E. S. Kuh: "Efficient Algorithms for Channel Routing". UCB/ERL M80/43, Aug. 1980.

[3] D. N. Deutsch: "A Dogleg Channel Router". Proc. 13th Design Auto. Conf. 1976, pp. 425-433.

[4] Y. K. Chen and M. L. Liu: "Three layer Channel Routing". UCB/ERL M81/84, Nov. 1981.

[5] M. L. Liu and Y. K. Chen: " Double-layer Channel Routing with Irregular Boundaries". UCB/ERL M81/79, Oct. 1981.

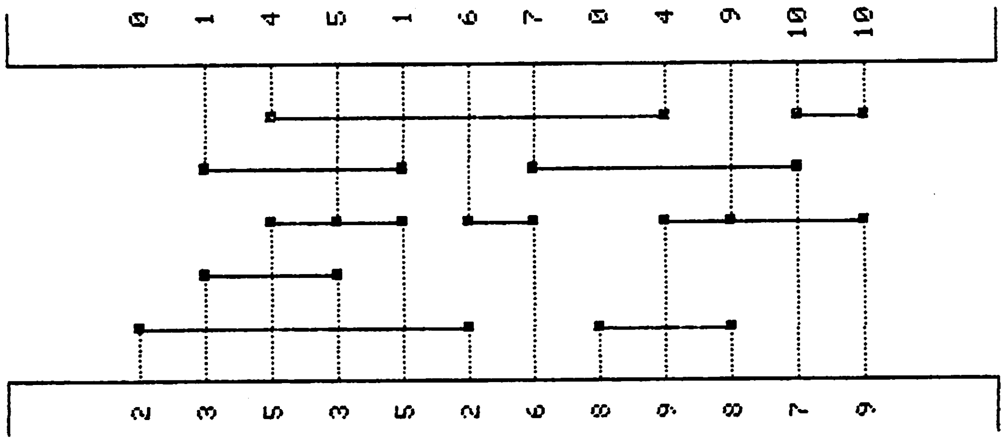


Fig. 1-1

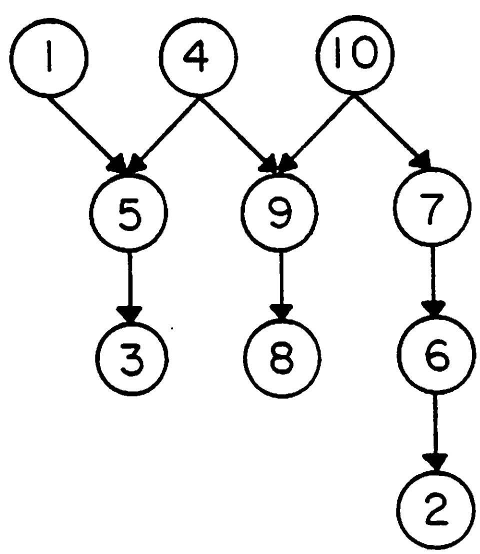


Fig. 1-2

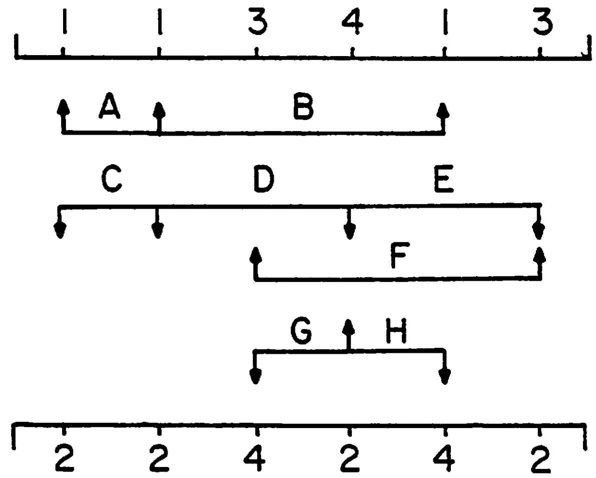


Fig. 1-3

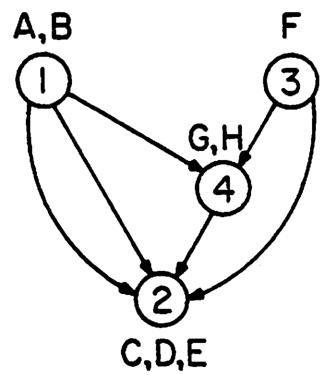


Fig. 1-4 (a) without dogleg

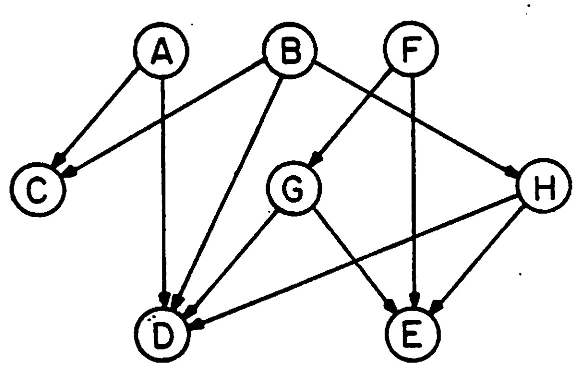


Fig. 1 4(b) with dogleg

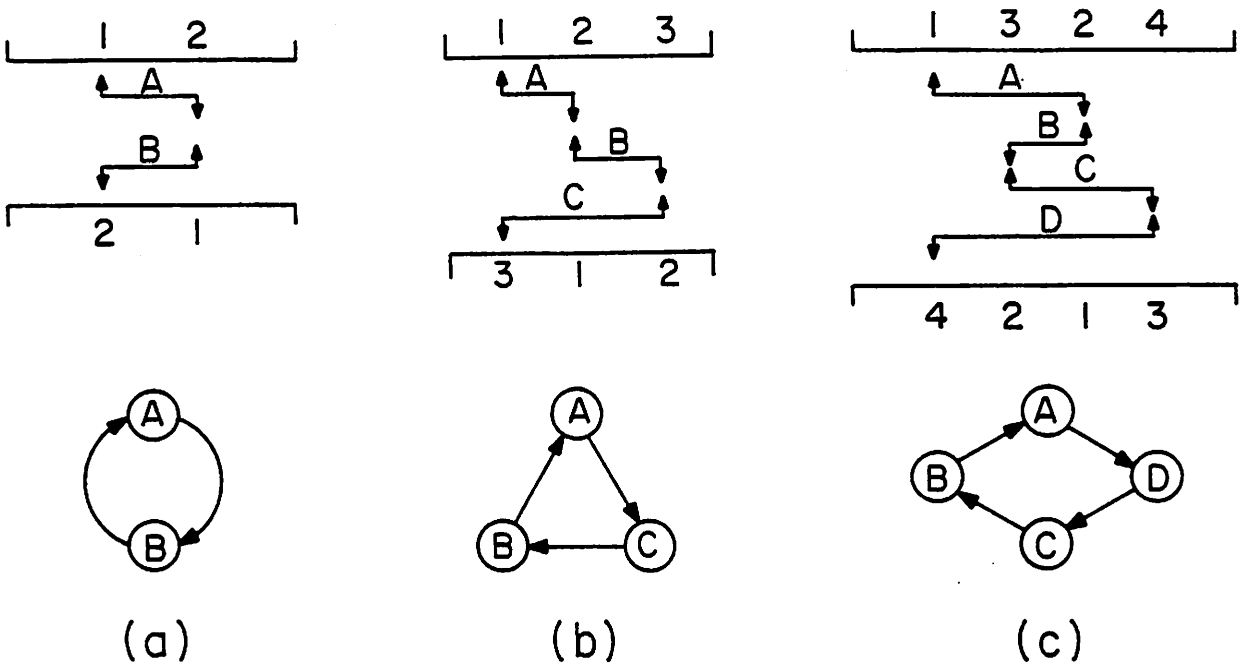

1 2
A
B
2 1
1 2 3
A
B
C
3 1 2
1 3 2 4
A
B
C
D
4 2 1 3
A
B
A
B
C
A
B
C
D
(a)
(b)
(c)

Fig. 1-5

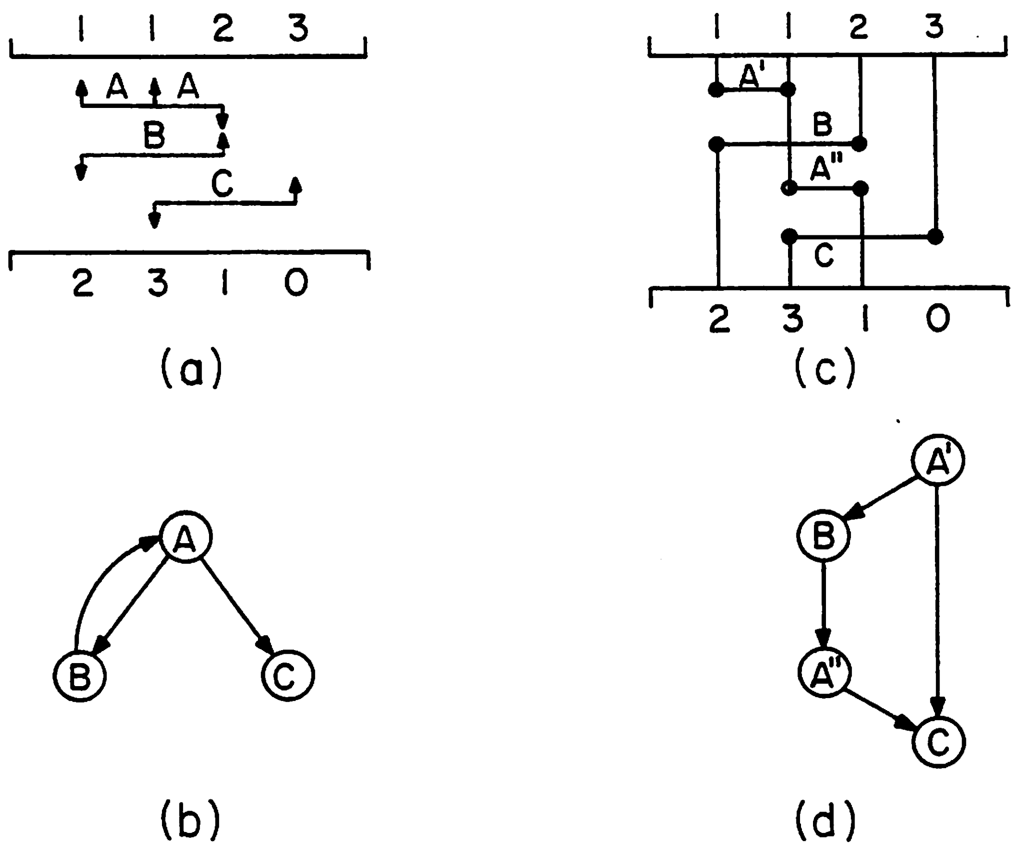

1 1 2 3
A A
B
C
2 3 1 0
(a)
1 1 2 3
A'
B
A"
C
2 3 1 0
(c)
A
B
C
(b)
A'
B
A"
C
(d)

Fig. 1-6

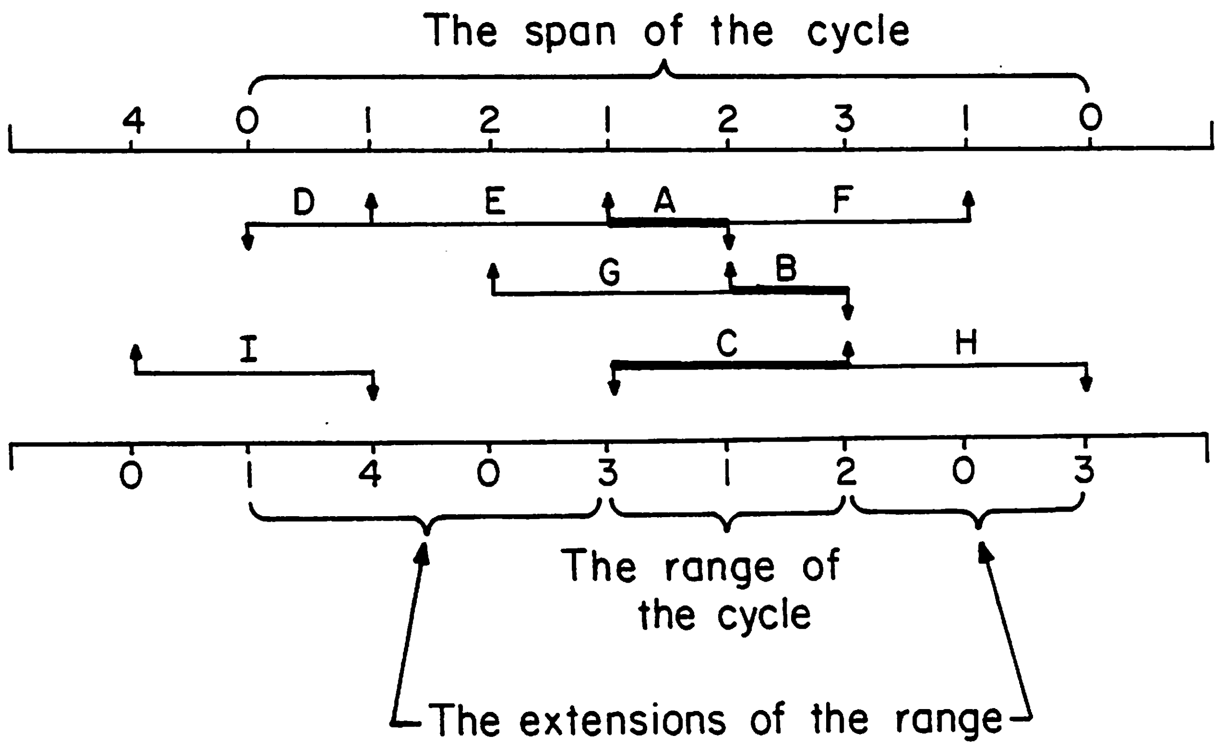


Fig. 2-1

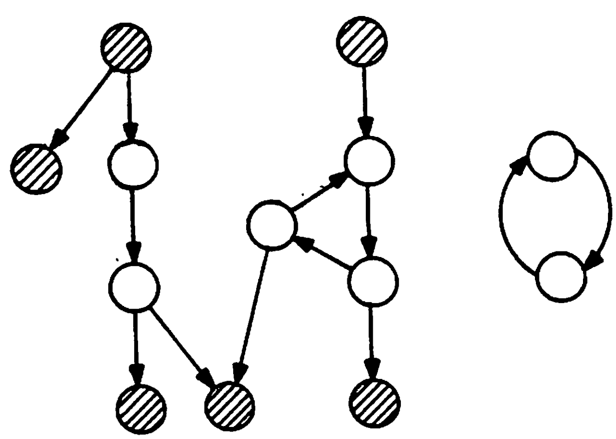


Fig. 2-2 exposed nodes

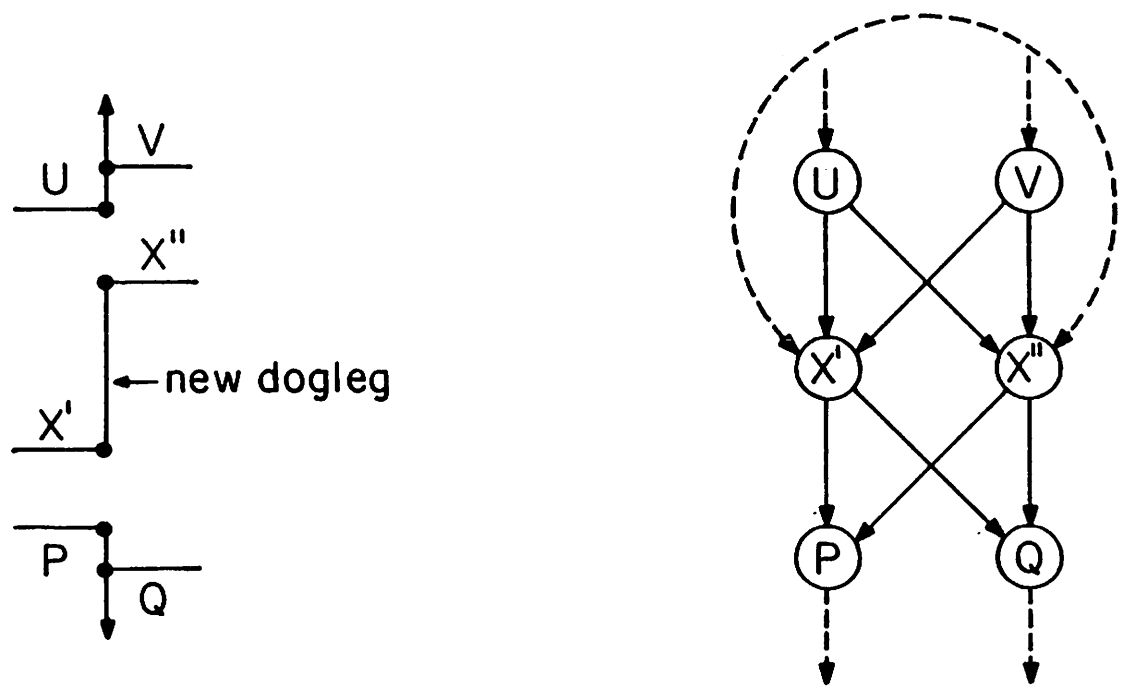
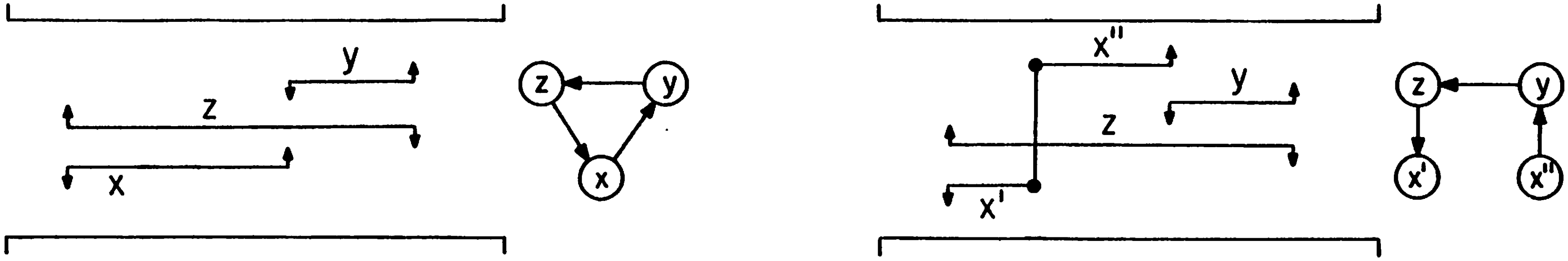
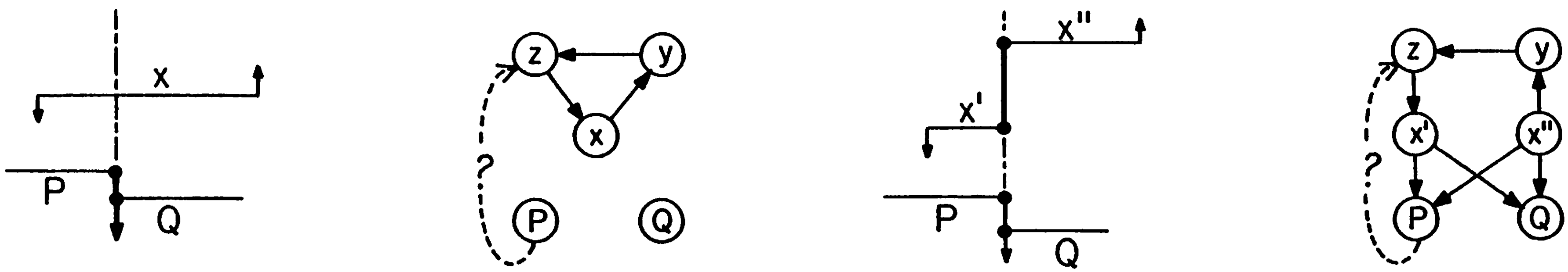


Fig. 3-1



(a) (b)

Fig. 4-1 method I



(a) before the breaking (b) after the breaking

Fig. 4-2

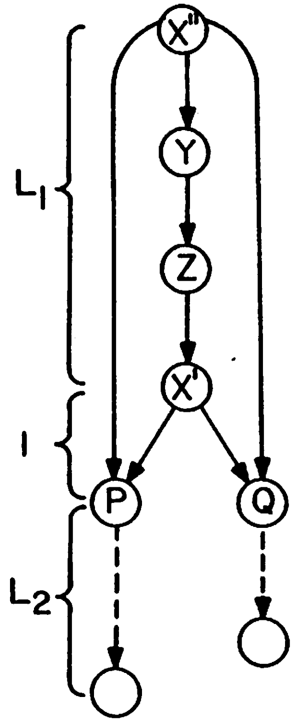


Fig. 4-3

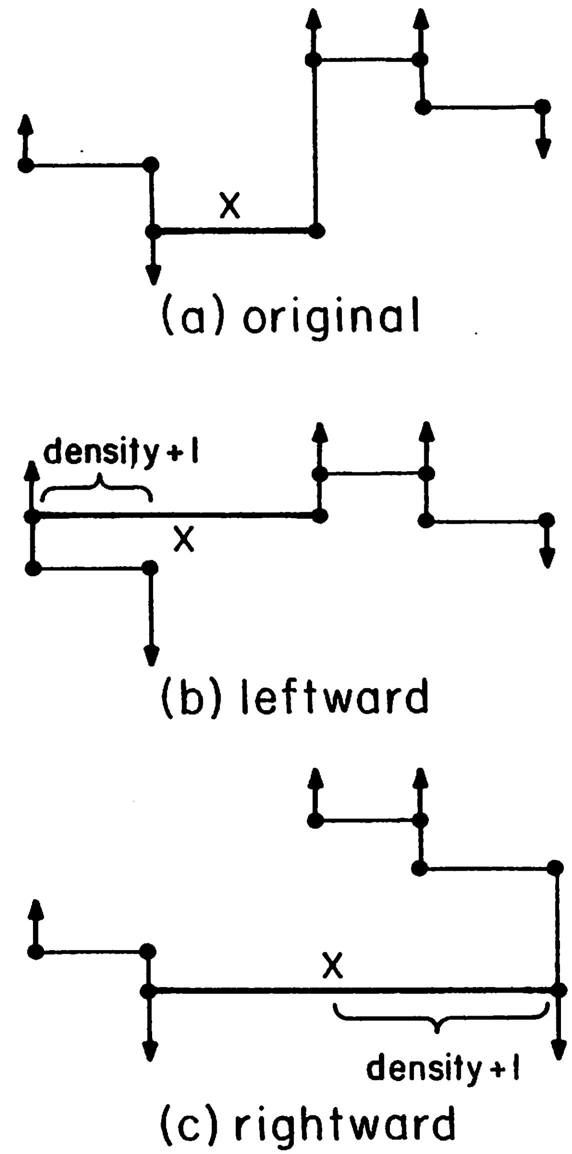
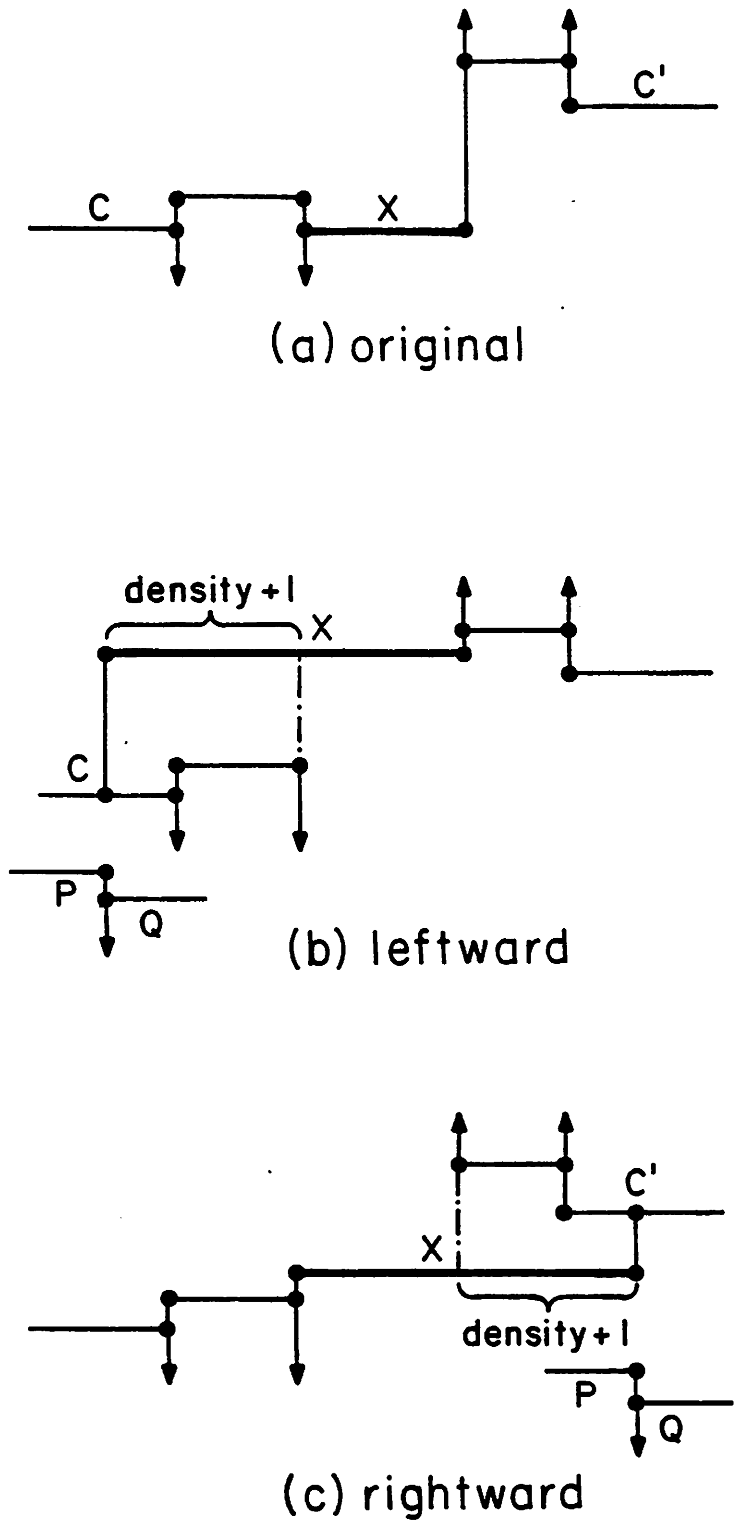


Fig. 5-1 method 2



(a) original
(b) leftward
(c) rightward

Fig. 6-1 method 3

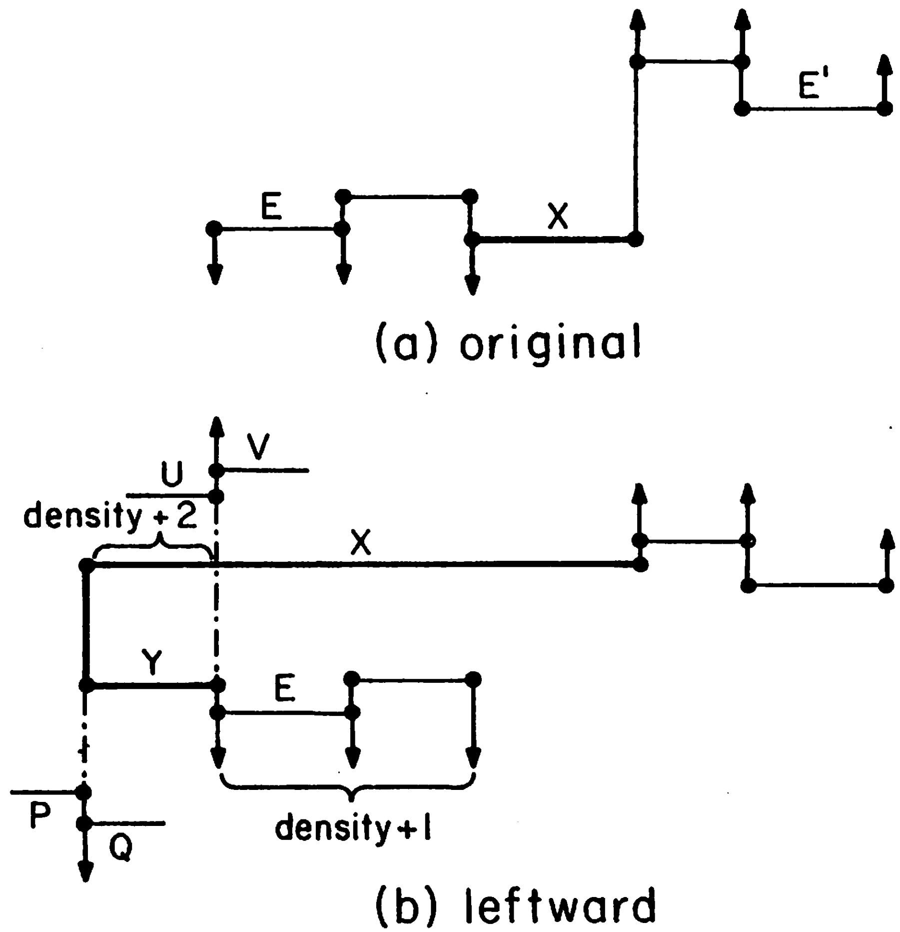


Fig. 7-1 method 4

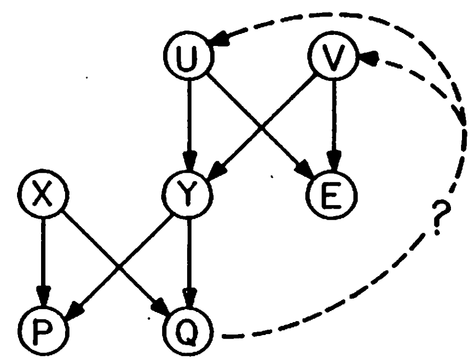


Fig. 7-2

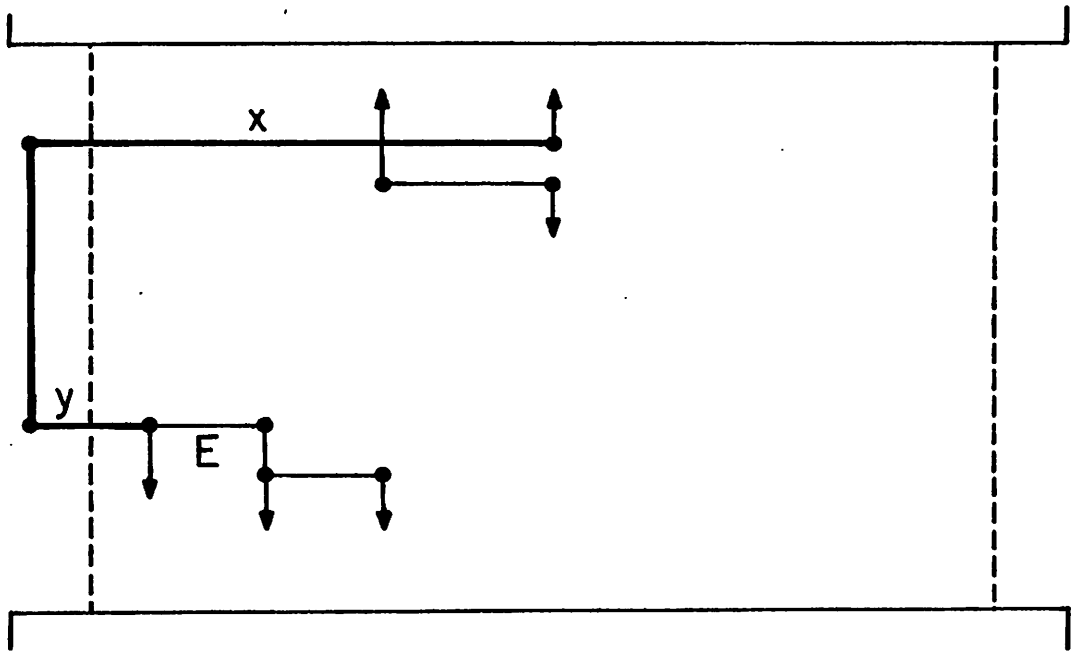
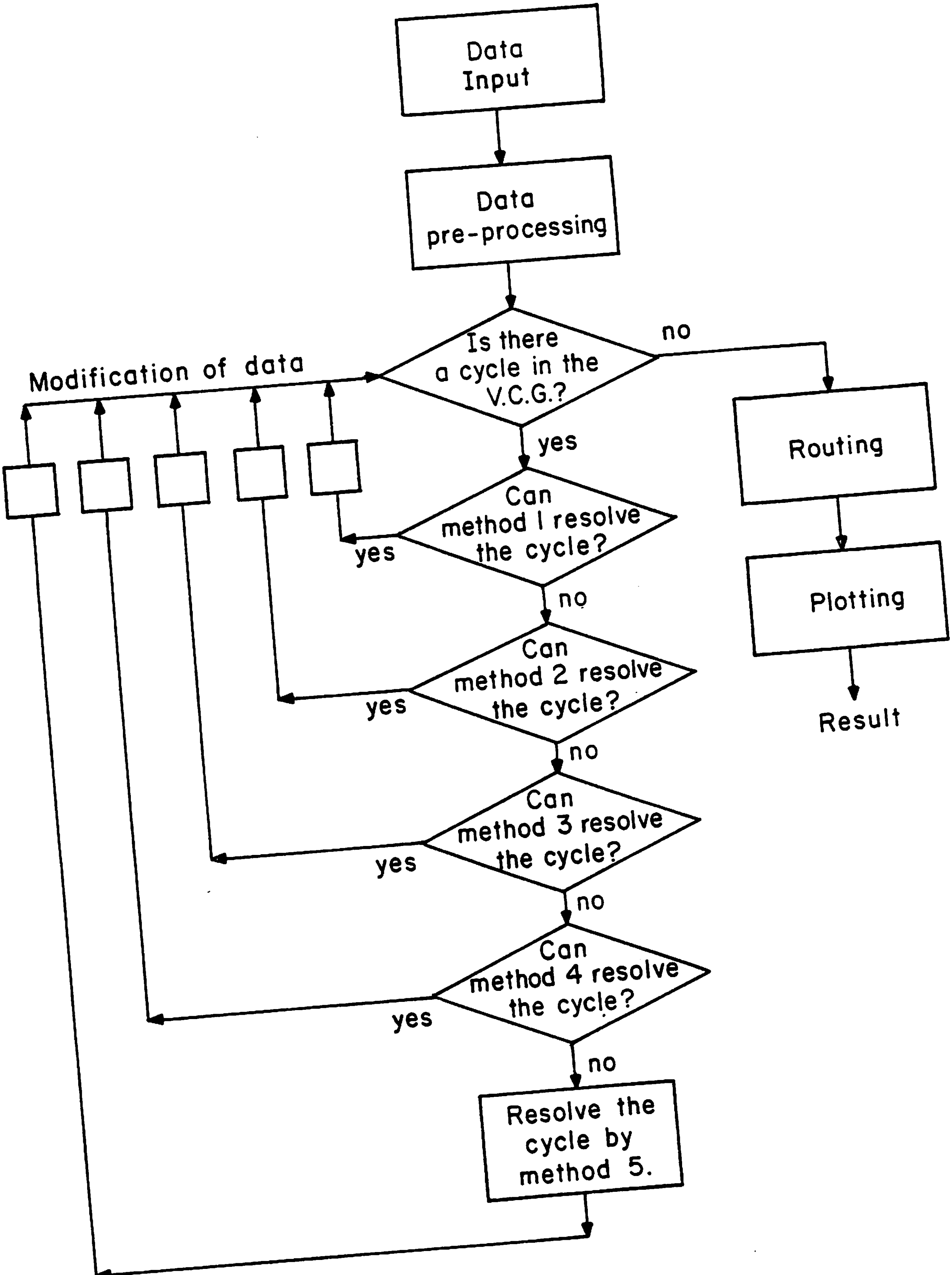


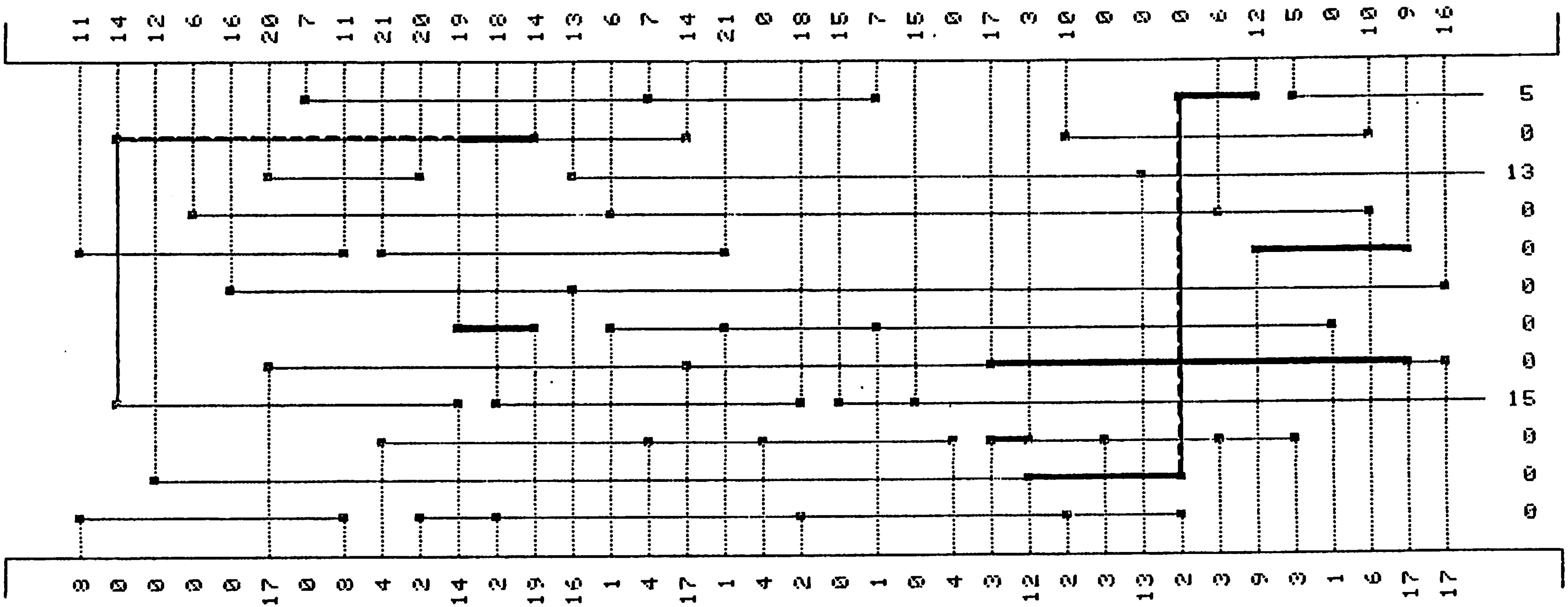
Fig. 8-1 method 5



Data Input
Data pre-processing
Is there a cycle in the V.C.G.?
no
Modification of data
yes
Routing
Can method 1 resolve the cycle?
yes
no
Plotting
Can method 2 resolve the cycle?
yes
Result
no
Can method 3 resolve the cycle?
yes
no
Can method 4 resolve the cycle?
yes
no
Resolve the cycle by method 5.

Fig. 10-1

**************example c7******************
(with dogleg)



\# column= 40   # zone= 24   before merging vmax= 6   maximum density= 12
\# net= 21   start zone= 8   after merging vmax= 6   number of tracks= 12

Fig. II-1 method 1 and method 2

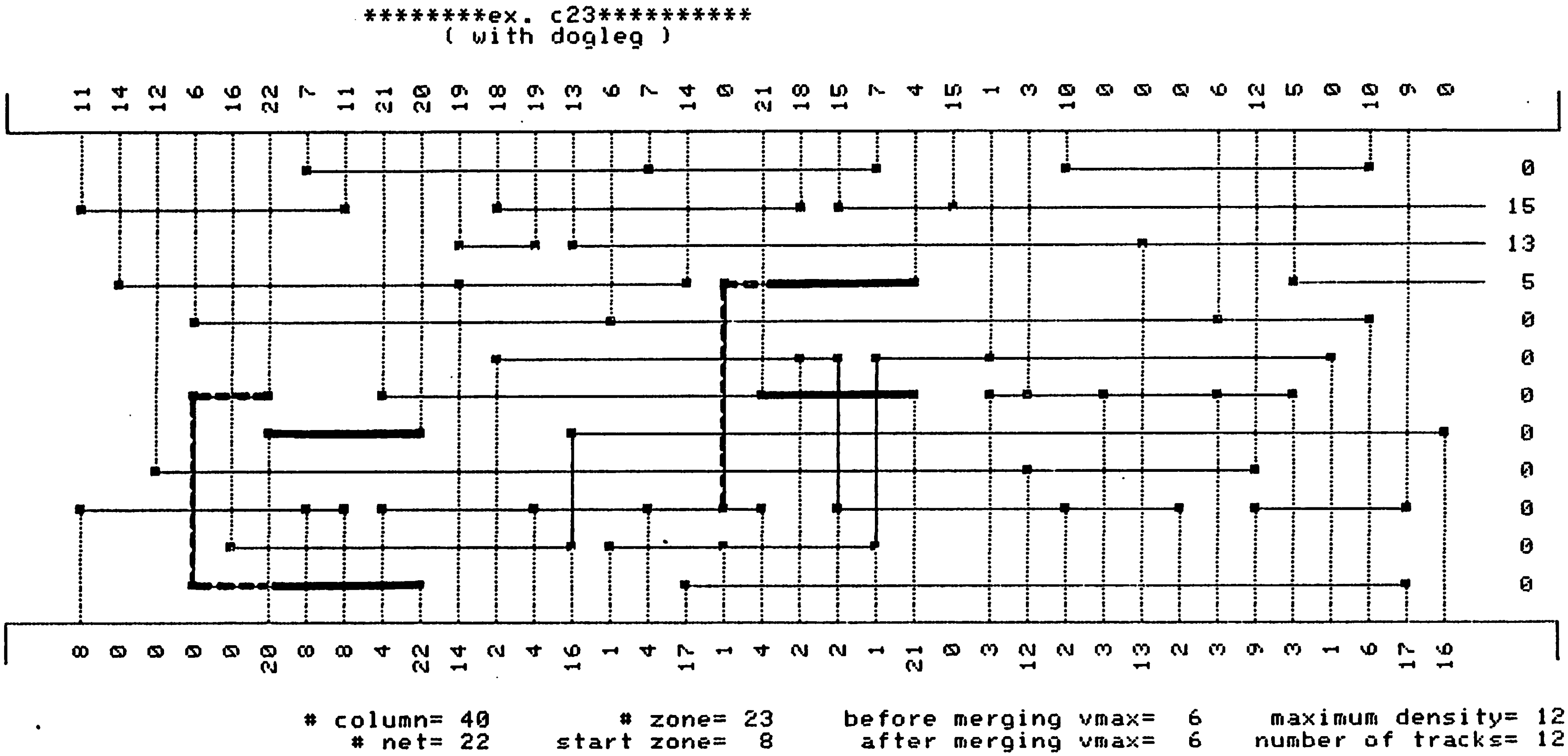
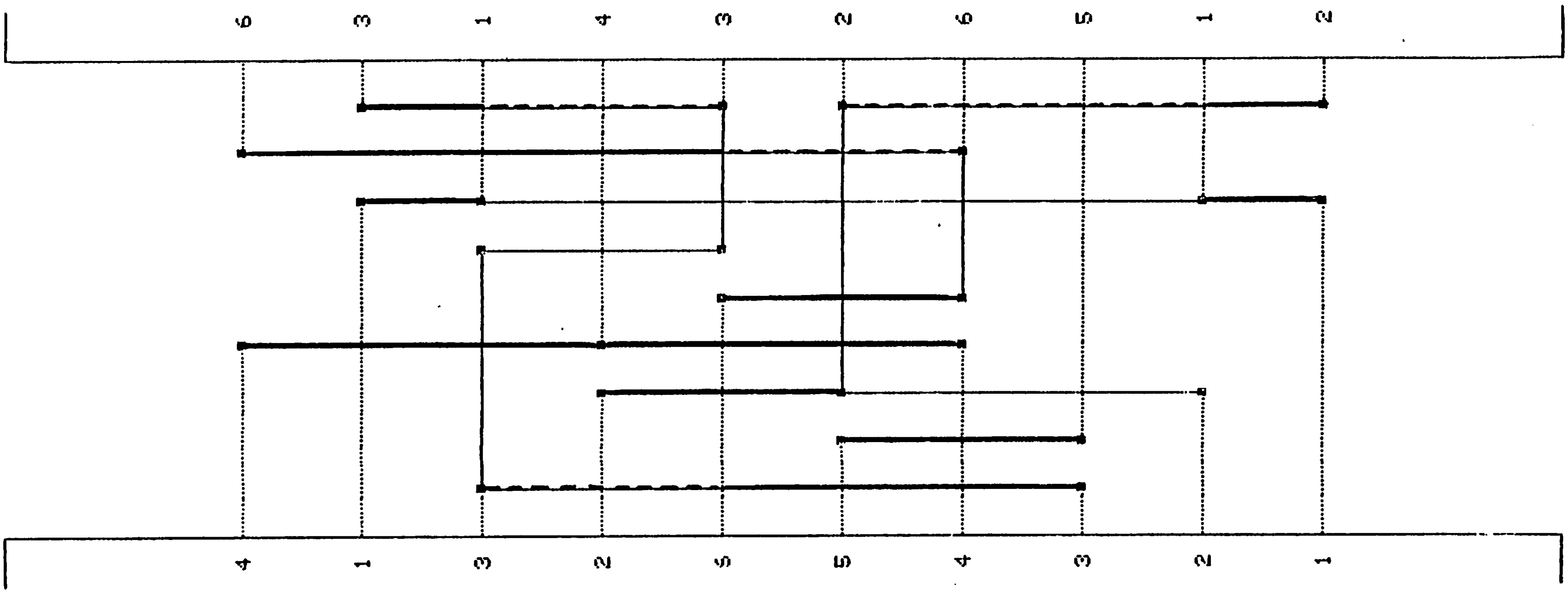


Fig. 11-2 method 3 and method 4

*********ex c14***********
(with dogleg)

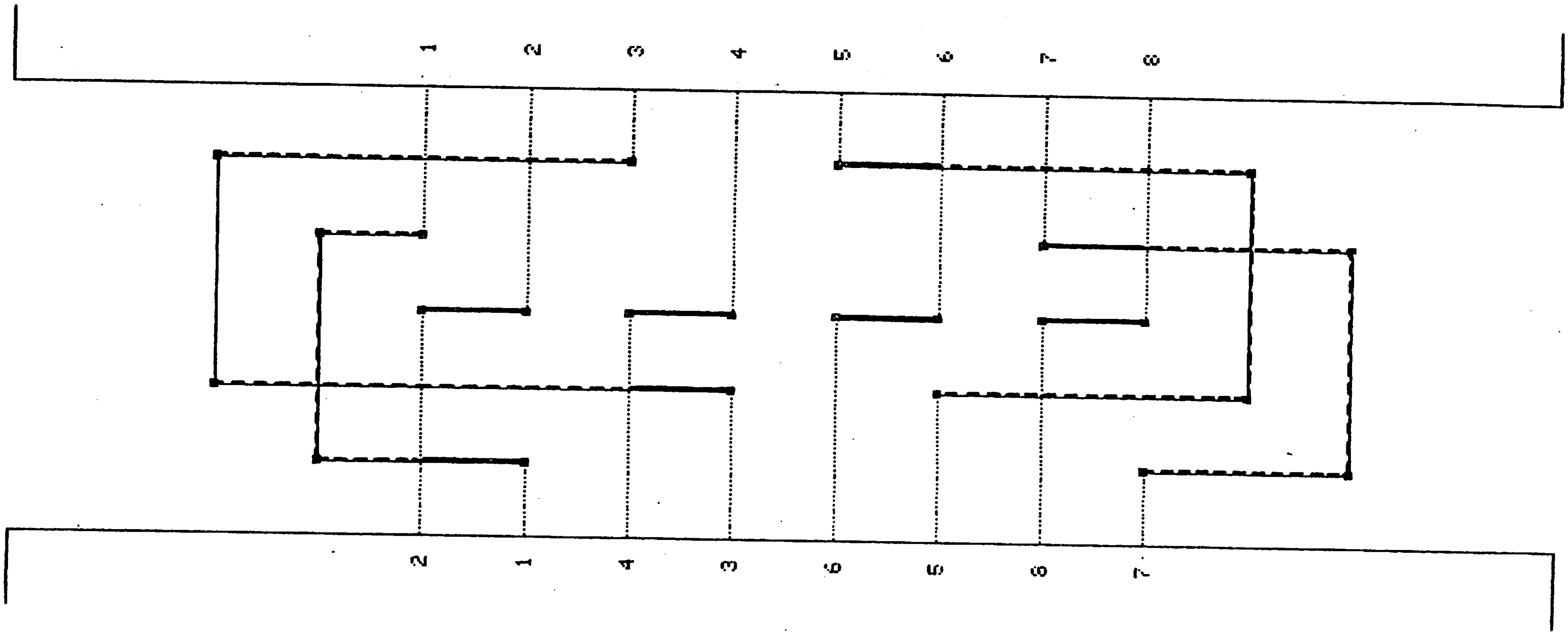


\# column= 10   # zone= 6   before merging vmax= 8   maximum density= 6

\# net= 6   start zone= 1   after merging vmax= 8   number of tracks= 9

Fig. II-3 (method 2)× 4

***********ex c19************
(with dogleg)



\# column= 8 # zone= 4 before merging vmax= 3 maximum density= 2
\# net= 8 start zone= 1 after merging vmax= 3 number of tracks= 5

Fig. II-4 (method 5)x4