Efficient Block Packing to Minimize Wire Length and Area

Katsumi Harashima Yousuke Ootaki Toshirou Kutsuwa

Osaka Institute of Technology
5-16-1 Omiya, Asahi-ku, Osaka, 535-8585 Japan
Tel: +81-6-6954-4305, Fax: +81-6-6957-2136
E-mail: ootaki@cs.elc.oit.ac.jp

Abstract: In layout of LSI and PWB, block packing problem is very important in order to reduce chip area. Sequence-pair is typical one of conventional packing method and can search nearly-optimal solution by using Simulated Annealing(SA). SA takes huge computation time due to evaluating of various packing results. Therefore, Sequence-pair is not effective enough for fast layout evaluation including estimation of wire length and rotation of every blocks. This paper proposes an efficient block packing method to minimize wire length and chip area. Our method searches an optimal packing efficiently by using a cluster growth algorithm with changing the most valuable packing score on packing process.

1. Introduction

Block Packing is an approach for layout of LSI and PWB. Sequence-pair is one of the most effective methods in Block Packing. In order to minimize layout area, Sequence-pair can search a global nearly-optimal solution with Simulated Annealing. It can also minimize the total wire length by adding evaluations of wire length to the objective function. However, it can not be applied to large packing problems in reasonable time owing to large computation time [1]–[3].

In this paper, we propose an efficient block packing method to minimize layout area and total wire length. Our method is based on a cluster growth algorithm, and can minimize total wire length and layout area efficiently by changing the most valuable packing score from wire length to layout area suitably on our packing process.

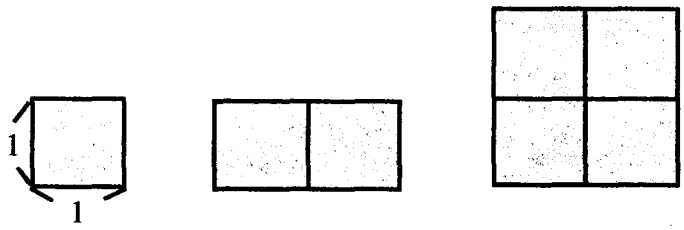


Figure 1. Layout blocks.

2. Layout Block and Area

In order to evaluate wire length and layout area concurrently for large problems, we simplify a packing problem.

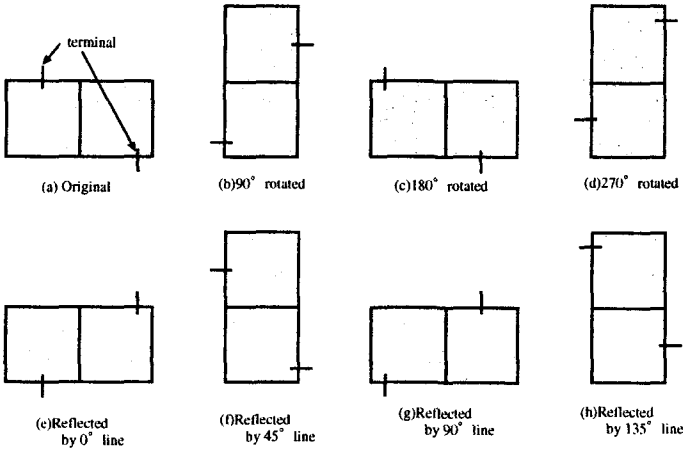


Figure 2. Rotations and reflections.

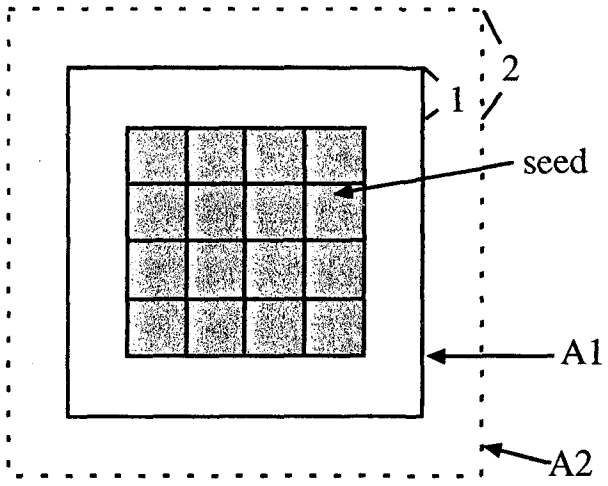


Figure 3. Placement area.

The proposed method takes only three types of blocks as shown in Fig.1. Type 1 in Fig.1(a) is an elementary block of which the length of each side is one. Type 2 and 3 are combinations of two and four elementary blocks respectively. Moreover, our method applies only rotations and reflections to every blocks as shown in Fig.2.

A1 shown in Fig.3 is a target placement area around the seed, the width of which is one. A2 shown in Fig.3 is a next target area around A1, the width of which is one. A2 is used by placements of type 2 or 3 blocks.

A1 and A2 placement area are divided many slots as shown in Fig.4. We call a slot or ones of which the figure is equal to a type 1 or 2 bounded by placed blocks in A1 or A2 a gap.

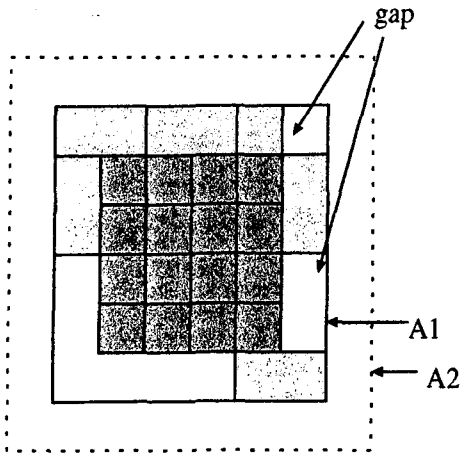


Figure 4. A gap.

3. Placement Algorithm

Our placement is based on a cluster growth algorithm and can search an optimal solution efficiently by deciding a rotation and reflection of a candidate block B after the decision of a place position with making a rough estimate of the wire length between B and the seed. The proposed method are organized as follows;

S1 Select the largest block among all ones taking the largest number of wiring requirements as the seed in order to estimate many wiring requirements.

S2 Choose a suitable candidate block Bi as will hereinafter be described in detail.

S3 Place Bi on A1 as shown in Fig.5. Assuming that each target terminal of every nets in the seed is located in the center of all terminals of each net, estimate wire length between Bi and target terminals in this phase. This phase consists of two steps.

S3-1 Give priorities to empty slots. A slot which the placement of Bi to it makes shorter wire length gets a higher priority.

S3-2 If A1 is not half full with placed blocks, Bi is placed on the placeable position with the highest priority. This placement is named WLF(wire length first). Otherwise, Bi is placed on the gap with the highest priority so as to approach the figure uniting the seed and Bi to a rectangle. This placement is named AF(area first).

S4 After deciding the place position, place Bi by evaluating rotations and reflections of Bi (Fig.2) to reduce wire length.

S6 If A1 is full and the total area of unplaced block is larger than the total area of empty slots in A2, change A2 the new target area A1.

S7 If unplaced blocks are left, goto S2, otherwise end.

In **S2** our block selection employs two approaches; WLF and AF selections.

For WLF, WLF selection chooses the largest block Bi among taking the largest number of wiring requirements for the seed.

IF feasible, AF selection chooses a block Bi of which the figure is equivalent to a gap with highest-priority and

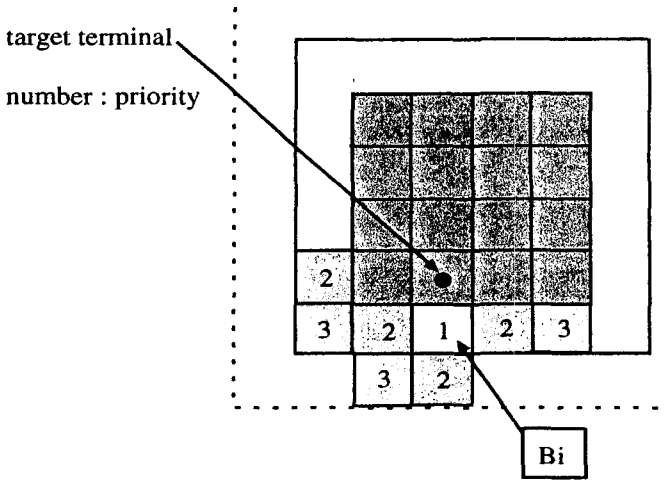


Figure 5. Priority of placement area.

which the wire length between Bi and the seed is shortest for WLF. Otherwise, AF selection chooses a block which takes the least wiring request for the seed.

4. Time Complexity

The worst-case time complexity of our method is $O(mn^2)$, where m is the number of wiring requirements, and n is the number of blocks. This can be shown as follows. Assume our method takes only type 1 blocks to consider the worst case. Our method consists of sorting and placement phases.

- In sorting phase, a candidate block is determined by search the largest number of wiring requirements for the seed. Suppose all blocks takes the m same wiring requirements. For determining a candidate block, evaluating the number of wiring requirements takes $O(m)$ for each n block, and thus determining a candidate takes $O(mn)$ (Fig.6). This process executes until no block is placed. Therefore, the amount of work for sorting phase is $O(mn^2)$.
- In placement phase, consider the packing probrem has n blocks and slots(Fig.7). Determination of a suitable slot for a first candidate block requires n evaluations of placement cost. In this way, the total number of evaluations is shown the following expression.

$$slots = n + (n-1) + (n-2) + \cdots + 1 \quad (1)$$

Consiquently, placement phase takes $O(n^2)$.

As described above, the worst-case complexity of our method becomes $O(mn^2)$.

5. Experiment

Figure8 shows experimental results. Figure8(a) is optimal. The results of (b) and (c) are based on our method. Figure8(b) only minimizes wire length, and (c) minimizes wire length under the restriction of final target placement area. Figure8(d) is ours. As the result of

n : the number of block

m : wiring requirements

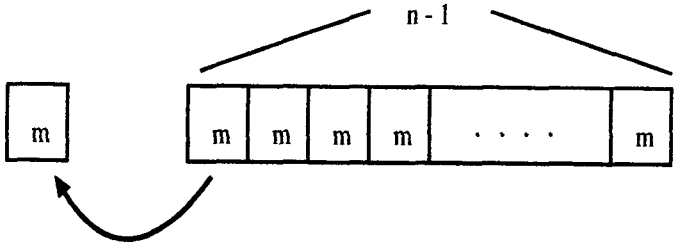


Figure 6. Decision of placement order.

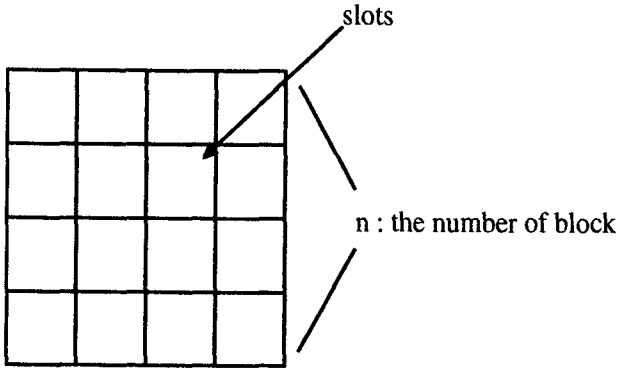


Figure 7. Search empty slot.

Fig.8(b), although minimization of wire length is easy, the total area has been larger than the optimal resutl. Given the area resutriction, the packing result is not effect as shown in Fig.8(c) because the total wire length has been increased to roughly four times than the optimal. On the contrary, our method can minimize wire length and layout area in a balanced manner.

In ours, the placements of block 12 and 13 in AF result in longer wire length. However, we consider that increase of wire length due to AF is little in large packing probrems.

6. Conclusion

We have proposed a packing method which considers wire length. Our method is based on a cluster growth algorithm, and can minimize total wire length and layout area efficiently by changing the most valuable packing score form wire length to layout area suitably on our packing process. Further, we showed that our method is efficient by the time complexity.

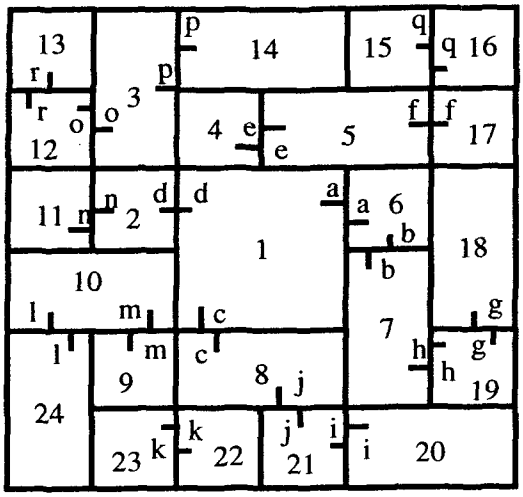
Our future works are to confirm of the effect of our method by a lot of experiments and to increase types of blocks.

References

[1] M. Hiroshi, K. Fujiyoshi, S. Nakatake and Y. Kajitani, "VLSI module placement based on rectangle-packing by the sequence-pair," IEEE Trans. Computer-Aided Design, vol. 15, pp. 1518-1524, 1996.

[2] M. Hiroshi, K. Fujiyoshi and M.Kaneko, "VLSI/PCB placement with obstacles based on sequence pair," IEEE Trans. Computer-Aided Design, vol. 17, pp. 60-68, 1998.

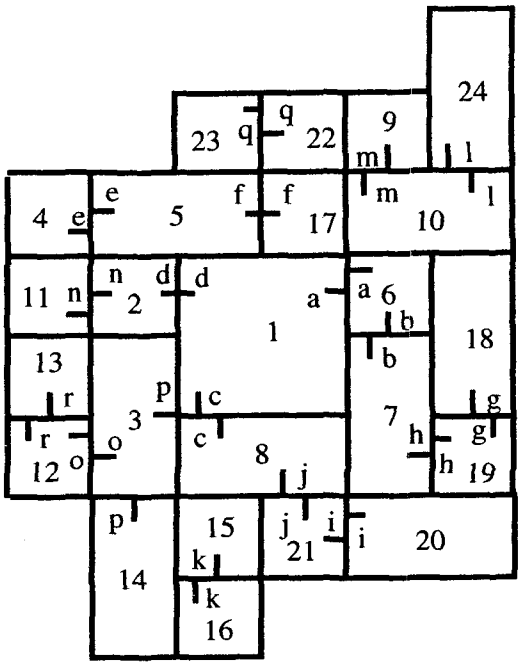
[3] X. Tang, R. Tian and D. F. Wong, "Fast evaluation of sequence pair in block placement by longest common subsequence computation," IEEE Trans. Computer-Aided Design, vol. 20, pp.1406-1413, 2001.



Total wire length : 4.25

Total area : 36

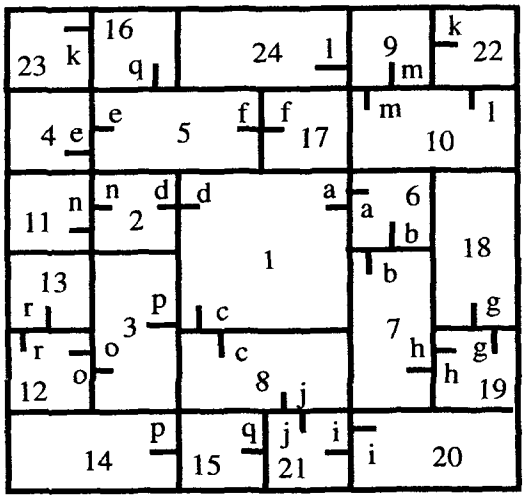
(a) Optimal result



Total wire length : 5.25

Total area : 48

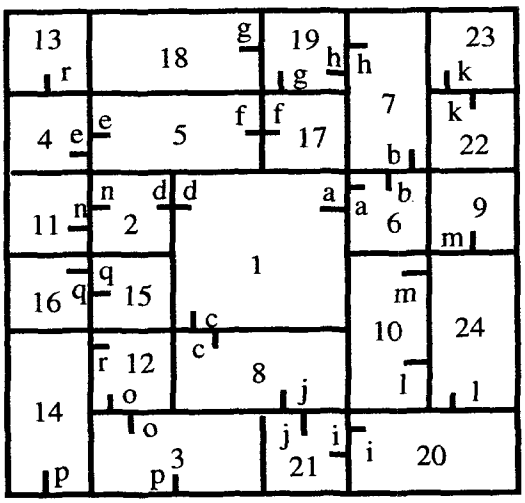
(b) Minimizing wire length



Total wire length : 16.00

Total area : 36

(c) Minimizing wire length and area



Total wire length : 10.75

Total area : 36

(d) Ours

Figure 8. Results.