

|              |   |
|--------------|---|
| Title        | 周期的繰り返し矩形配置の解表現と配置最適化   |
| Author(s)    | 小川, 智之  |
| Citation     |   |
| Issue Date   | 2005-03   |
| Type         | Thesis or Dissertation  |
| Text version | author  |
| URL          | <a href="http://hdl.handle.net/10119/1929">http://hdl.handle.net/10119/1929</a> |
| Rights       |   |
| Description  | Supervisor:金子 峰雄, 情報科学研究科, 修士   |

# A Finite Solution Space for Recurrent Placements of Rectangles

Tomoyuki Ogawa (310025)

School of Information Science,  
Japan Advanced Institute of Science and Technology

February 10, 2005

**Keywords:** floorplan, sequence pair, non-slicing placement.

The placement of rectangles is the problem to find the position of rectangles on a two-dimensional plane such that every pair of rectangles have no overlap. This problem is an optimization problem, which has various applications such as VLSI layout design. Despite of its simple appearance, many placement problems with different objective are known to be NP-hard, and approximation algorithms and stochastic algorithms have been developed.

In this research, we consider the problem to place multiple copies of a basic set of rectangles so that the placement of one basic set of rectangles repeats spatially and regularly on a plane. Such recurrent placement is motivated from a layout of regular circuit structure(array of homogeneous subcircuits), such as memory, multiprocessor on chip, FPGA(which has a regular structure of Basic Logic Cells, switch-boxes, and wire segments), etc. Placement of rectangles on a side wall of a cylinder and loop pipeline task schedule are also candidate applications of recurrent placement.

This research treats recurrent placement repeated only in  $x$  direction, as the first trial to recurrent placement, and the objective of this research is to propose a coding scheme for those recurrent placement and placement optimization.

“Sequence Pair” is known as such a coding system for conventional one-time placement. The sequence pair consists of two permutaions of rectangle

names, which is conventionally represented as  $(\Gamma_+, \Gamma_-)$ , and provides spatial relation of every pair of rectangles without any inconsistencies. The “gridding” is an algorithm which extracts sequence pair from a given placement (that is, encoding). The gridding is applicable to our recurrent placement, and sequences of rectangle names  $\Gamma_+$  and  $\Gamma_-$  extracted from a recurrent placement become cyclic. On the other hand, when we try to derive a recurrent placement from two cyclic sequences  $\Gamma_+$  and  $\Gamma_-$  (that is, decoding), there remains a great ambiguity in specifying spatial relations between rectangles, because a pair of  $\Gamma_+$  and  $\Gamma_-$  represents spatial relation of rectangles with confusing intra-cycle relation and inter-cycle relation.

To solve this problem in decoding, we introduced the third sequence  $\Gamma_s$  in addition to  $\Gamma_+$  and  $\Gamma_-$ , and developed a new coding scheme for the recurrent placement. We call the coding scheme “Sequence Triple” following after “sequence pair”. In the sequence triple, the third sequence  $\Gamma_s$  is introduced in order to specify the spatial order of rectangles in one cycle placement relevant to the sequence  $\Gamma_-$ . Compared with the sequence pair, the sequence triple uses one more permutation, sequence triple is able to specify intra-cycle spatial relation and inter-cycle spatial relation without confusion. The complexity of the decoding algorithm developed in this research is  $O(n^3)$ , and the size of solution space is  $O((n!)^3)$ , where  $n$  is the number of rectangles in one cycle.

We implemented a recurrent placement optimizer which is based on simulated annealing search of solution space given by the sequence triple using C program language. The input of the optimizer is the width and the height of rectangles, and the output is the coordinates of rectangles. The objective is the minimization of “cycle width  $L \times$  total height  $H$ ”. We experimented with various input instances, and we have recognized that the optimizer tends to generate a taller placement. This tendency is supported also by the statistical analysis of sequence triple. That is, the probability of two rectangles to be in above-below relation is somewhat larger than the one of two rectangles to be in left-right relation. Hence, to find a better solution by the SA search of the solution space, some device to cancel such bias of probabilities would be needed.

The following remains as future works, (1) choice of proper set of generation rules for generating neighbor solution, (2) development of decoding

algorithm to speed up decoding task, and (3) development of other coding scheme superior to “sequence triple” in the computational complexity and the size of solution space.