

Dimension of Partial Orders and Its Application to Rectangle Packing

宮下 弘 梶谷洋司

Hiroshi Miyashita Yoji Kajitani

北九州市立大学 国際環境工学部 情報メディア工学科

Department of Information and Media Sciences

Faculty of Environmental Engineering, The University of Kitakyushu

Abstract

A sequence pair can be used to represent a set of instances of a rectangle-packing problem so that the set is guaranteed to include the minimum-area layout. The sequence pair has attracted much attention from CAD researchers because it provides them with an efficient representation of the instances of the rectangle-packing problem without restricting layouts to slicing structure. In this respect, the sequence pair is the first representation that can adequately handle realistic requirements. This paper relates the sequence pair to the notion of dimension of partial orders and clarifies its mathematical background.

1 Introduction

The notion of the sequence pair was proposed in [1]. A sequence pair can be used to represent feasible instances of the rectangle-packing problem in a simple form. The rectangle-packing problem is one of the fundamental problems in LSI layouts, and its purpose is to pack a given set of rectangular modules without overlapping so that the chip area can be minimized. Since the publication of the paper, the sequence pair has attracted much attention among CAD researchers because it is the first efficient representation that can be applied to instances that do not necessarily have a slicing structure.

The sequence pair is a pair of sequences whose terms are the names of rectangular modules placed on a chip. It is characterized by the feasible placement of the rectangular modules. This means that the feasible placement of modules can be determined by the sequence pair and, conversely, that a sequence pair can be constructed from the feasible placement of modules. The solution space defined by the sequence pair not only can be easily enumerated but can also be confirmed to include the optimum solution of the minimum area. Hence, if we construct a

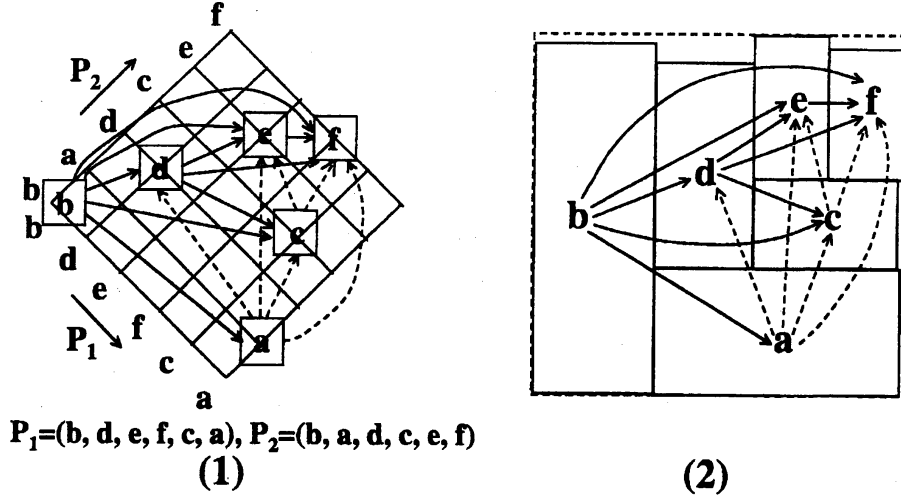


Figure 1: Sequence pair representing rectangle placement. (1) Sequence pair. (2) Rectangle packing.

mechanism such as simulated annealing to search for the instances in the solution space, we can obtain a better solution than those by previously proposed methods as long as sufficient computing time is given.

For these reasons, the sequence pair continues to be used in the placement stage of VLSI (Very large scale integrated circuits) layout design process. However, the mathematical background of the sequence pair has not been clarified yet. In this paper, we relate the sequence pair to the notion of dimension of partial orders and clarify its mathematical background.

The rest of this paper is organized as follows. In Section 2, we review the notion of the sequence pair. We introduce the dimension of partial orders in Section 3. Section 4 relates the sequence pair to the dimension of partial orders. The rectangle packing method based on simulated annealing and some application results are described in Section 5. Finally, Section 6 gives some concluding remarks.

2 Sequence Pair

Figure 1 shows an example of the sequence pair. Figure 1(1) demonstrates how to represent the rectangle placement described in (2) by a sequence pair, $P_1 = (b, d, e, f, c, a)$ and $P_2 = (b, a, d, c, e, f)$. A solid (broken) arrow represents horizontal (vertical) relation between rectangles. The P_1 sequence corresponds to the axis having slope -1 , while the P_2 sequence corresponds to the axis having slope $+1$. We consider an orthogonal coordinate system that is

determined by the two axes above. The coordinates are confined to $n \times n$ grid lines separated uniformly, where n is the number of rectangular modules. Each grid line corresponds to a rectangle in the same order as it appears in the sequence. Each module is placed on the grid point forming the intersection of the two orthogonal grid lines determined by the rectangle.

Assume an xy orthogonal coordinate system on the plane where a set of rectangles are placed without overlapping one another. A rectangle that is placed above (right on) another rectangle implies that the former is included in the halfspace of $\{(x, y) | y \geq k\}$ ($\{(x, y) | x \geq k\}$), while the latter is in $\{(x, y) | y \leq k\}$ ($\{(x, y) | x \leq k\}$), for some constant k . On the basis of this implication, the correspondence between the rectangle placement and its sequence pair representation is defined as

$$P_1 = (\dots, i, \dots, j, \dots) \text{ and } P_2 = (\dots, i, \dots, j, \dots) \\ \xLeftrightarrow{\text{def}} \text{Rectangle } j \text{ is right on rectangle } i, \quad (1)$$

$$P_1 = (\dots, j, \dots, i, \dots) \text{ and } P_2 = (\dots, i, \dots, j, \dots) \\ \xLeftrightarrow{\text{def}} \text{Rectangle } j \text{ is above rectangle } i. \quad (2)$$

Given a rectangle packing, we can obtain a sequence pair that corresponds to the rectangle packing as demonstrated in Fig. 2. We move the rectangles slightly so that every two rectangles is separated each other. For each rectangle i , we draw lines as follows. First, the starting point, from which we begin to draw lines, is located at the upper right corner of i . Starting to move upward, we turn its direction alternately right and up until we reach the upper right corner without crossing: i) boundaries of other rectangles, ii) previously drawn lines, and iii) the boundary of the chip. The drawn line is called the up-right step-line of rectangle i . The down-left step-line is also drawn in a similar fashion. The union of these step lines together with the connecting diagonal line of rectangle i is called the positive step-line of rectangle i . We can draw such a positive step-line for each rectangle. The positive step-lines are referred to by the corresponding rectangles. An example of resultant positive step-lines is shown in Fig. 2(1). Since no two positive step-lines cross each other, they are linearly ordered. Selecting positive step-lines from left to right, we can obtain the sequence of rectangles $P_1 = (b, d, e, f, c, a)$.

Negative step-lines are drawn in a similar manner to the positive step-lines, as demonstrated in Fig. 2(2). The difference is that a negative step-line is the union of the left-up step-line and right-down step-line, whose directions alternate between left and up and between right and down, respectively. Ordering the negative step-lines also from left to right, we can reach the sequence of rectangles $P_2 = (b, a, d, c, e, f)$. Consequently, we can obtain a pair of sequences

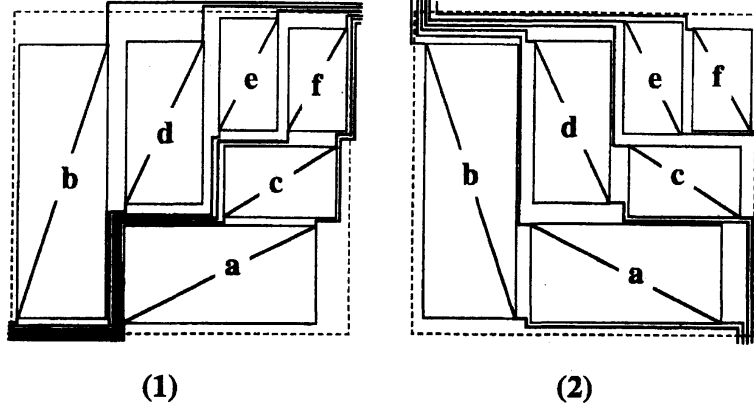


Figure 2: Conversion from a rectangle packing to the corresponding sequence pair. (1) Positive step-lines. (2) Negative step-lines.

P_1 and P_2 from a given rectangle packing.

Conversely, given a sequence pair, we can construct a rectangle packing that corresponds to the sequence pair. First, for horizontal relation between rectangles, we construct a directed and vertex-weighted graph called the horizontal-constraint graph $G_x(V, E_x)$ as shown in Fig. 3(1). Here, vertex set V comprises source vertex s , sink vertex t , and vertices labeled with rectangle names. For source s and sink t , there exist directed edges $(s, i) \in E_x$ and $(i, t) \in E_x$ for each rectangle i . There exists a directed edge $(i, j) \in E_x$ if and only if rectangle j is right on rectangle i as defined in equation (1). It should be remarked that the transitive directed-edges are omitted for simplicity in Fig. 3. Vertex-weight is defined as zero for source s and sink t , while width of rectangle i for the corresponding vertex i . Second, the vertical-constraint graph $G_y(V, E_y)$ is constructed using vertical relations defined as in equation (2) and the heights of rectangles in a similar fashion. See Fig. 3(2). These constraint graphs contain no directed cycles. Finally, after x - and y -coordinates of source vertex s are initialized by zero, longest path length from source s to vertex i is calculated in both $G_x(V, E_x)$ and $G_y(V, E_y)$ independently. For example, we can apply a longest path algorithm proposed in [2]. We set the x - and the y -coordinates of rectangle i to the longest path lengths from source vertex s to vertex i in $G_x(V, E_x)$ and $G_y(V, E_y)$, respectively. Here, x - and y -coordinates are the ones of the lower left corner of the

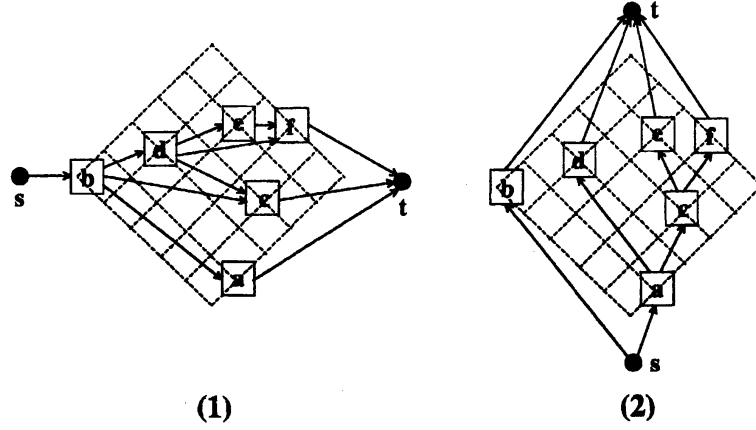


Figure 3: Conversion from a sequence pair to the corresponding rectangle packing. (1) Horizontal-constraint graph. (2) Vertical-constraint graph.

3 Dimension of Partial Orders

A partially ordered set (poset) is defined as a pair (X, P) where X is a set (finite in this paper) and $P \subset X \times X$ is a partial order on X . A partial order P is a binary relation on X that satisfies the following three conditions:

- (1) For all $x \in X$, $(x, x) \in P$. (reflexivity)
- (2) If $(x, y) \in P$ and $(y, x) \in P$, then $x = y$. (antisymmetry)
- (3) If $(x, y) \in P$ and $(y, z) \in P$, then $(x, z) \in P$. (transitivity)

The notations $(x, y) \in P$, $x \leq y$ and $y \geq x$ in P are used interchangeably. If $(x, y) \in P$ and $x \neq y$, then we use the obvious notation $x < y$ or $y > x$ in P . Elements x, y in X are said to be comparable if (x, y) or (y, x) in P ; otherwise x and y are incomparable, which is denoted by $x \parallel y$. Let $\mathcal{I}_P = \{\{x, y\} \mid x, y \in X, x \parallel y \text{ in } P\}$. If $\mathcal{I}_P = \emptyset$, then P is called a linear order on X and (X, P) is called a linearly ordered set or a chain.

If P and Q are partial orders defined on X such that $P \subset Q$, then Q is said to be an extension of P . In particular, if $P \subset Q$ and Q is a linear order, we call Q a linear extension of P .

For any binary relation R on X , the transitive closure $\bar{R}$ of R is a set of $(x, y) \in X \times X$ for which there exists a sequence $x_1, x_2, \dots, x_n \in X$ such that $n \geq 2$, $x_1 = x$, $x_n = y$, and $(x_i, x_{i+1}) \in R$ for every i ($1 \leq i \leq n-1$).

For a given poset (X, P) , we can construct a directed graph $G_P = (V, E)$ as follows. The vertex set V is defined by $V \stackrel{\text{def}}{=} X$ and the directed-edge set E by $(x, y) \in E$ if and only if $x < y$ in P . Since $(x, y) \in E$ and $(y, z) \in E$ imply $(x, z) \in E$, the transitivity of directed edges

holds. An undirected graph $G = (V, E)$ is transitively orientable if each edge can be directed so that the transitivity holds. A comparability graph is an undirected graph that is transitively orientable. For an undirected graph $G = (V, E)$, its complementary graph $G^c = (V, E^c)$ is an undirected graph defined by $\{x, y\} \in E^c$ if and only if $\{x, y\} \notin E$.

Lemma 1 *If P is a partial order on X and $\{a, b\} \in \mathcal{I}_P$, then $\overline{P \cup \{(a, b)\}}$ is a partial order on X .*

Proof: Reflexivity and transitivity are obviously satisfied. Suppose that $x \geq y$ and $y \geq x$ for $x, y \in X$. From the definition of the transitive closure, there exists a directed loop $x_1, x_2, \dots, x_n$ such that $(x_i, x_{i+1}) \in P \cup \{(a, b)\}$, $(1 \leq i \leq n)$ and $x_{n+1} = x_1$, which includes x and y . If the sequence does not include (a, b) , then $x = y$. Otherwise, a and b become comparable in P , which is a contradiction (i.e., $\{a, b\} \notin \mathcal{I}_P$). Hence, antisymmetry also holds. Thus, $\overline{P \cup \{(a, b)\}}$ is a partial order on X . ■

Given a reflexive binary relation whose graph representation is $G = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$, we can generate its transitive closure by applying Warshall's algorithm to G [3]. The time complexity of the algorithm is $O(n^3)$.

Theorem 1 *If P is a partial order on X , then the collection $\mathcal{C}$ of all linear extensions of P is nonempty and $\cap \mathcal{C} = P$.*

Proof: If $\mathcal{I}_P \neq \emptyset$, then we can choose $\{a, b\} \in \mathcal{I}_P$. From Lemma 1, we can construct a partial order $P_1 = \overline{P \cup \{(a, b)\}}$.

If P_1 is not a linear order, then we again choose another $\{c, d\} \in \mathcal{I}_{P_1}$, which remains incomparable, and construct a partial order $P_2 = \overline{P_1 \cup \{(c, d)\}}$. This procedure is repeated until we reach a linear order $P_{(a,b)}$, which depends on the first selection (a, b) . Thus, $\mathcal{C}$ is not empty. For any $\{x, y\} \in \mathcal{I}_P$, we can construct linear orders $P_{(x,y)}$ and $P_{(y,x)}$ as described above. Let $\mathcal{C}$ be the set of all linear orders constructed in this way. Clearly, $\cap \mathcal{C} = P$ holds. ■

Let F be an edge orientation of the complete graph K_n on n vertices. The set F is a transitive tournament if and only if $(x, y) \in F$ and $(y, z) \in F$ imply $(x, z) \in F$. Clearly, this is equivalent to the condition that there exists no 3-cycle. It should also be noted that a linear order precisely corresponds to a transitive tournament. Thus we can consider the theorem below as a characterization of linearly ordered sets or chains. See [4] for detailed proof.

Theorem 2 [4] *Let F be an edge orientation of the complete graph K_n . The following statements are equivalent.*

- (1) F is a transitive tournament.
- (2) F is acyclic.
- (3) The vertices can be linearly ordered $(v_1, v_2, \dots, v_n)$ such that v_i has in-degree $i - 1$ in F for all $i = 1, 2, \dots, n$.
- (4) The vertices can be linearly ordered $(v_1, v_2, \dots, v_n)$ such that $(v_i, v_j) \in F$ if and only if $i < j$.

From Theorem 2, we can consider a linearly ordered set (chain) as a sorted sequence of its elements. Thus we can apply topological sort to construct a collection $\mathcal{C}$ of linear extensions directly in Theorem 1.

Let $G = (V, E)$ with $V = \{v_1, v_2, \dots, v_n\}$ be a graph representation of a partial order P on X . If $\{x, y\} \in \mathcal{I}_P$, then $G = (V, E \cup \{(x, y)\})$ is acyclic. By applying topological sort based on the depth-first search to graph G , we can obtain a linear extension. As in the proof of Theorem 1, a collection $\mathcal{C}$ of linear extensions can be generated by applying topological sort to edge sets $E \cup \{(x, y)\}$ and $E \cup \{(y, x)\}$ for all $\{x, y\} \in \mathcal{I}_P$. This collection of linear extensions $\mathcal{C}$ satisfies $\cap \mathcal{C} = P$. While the time complexity needed to generate transitive closure of G is $O(n^3)$, the time complexity of depth-first search is $O(m + n)$ when $|E| = m$. Thus direct application of topological sort is superior to that of transitive closure.

Definition 1 For a partially ordered set (X, P) , its dimension denoted by $\dim(X, P)$ is the smallest positive integer m such that there exists a set of linear extensions $\{L_1, L_2, \dots, L_m\}$ that satisfies $\cap_{i=1}^m L_i = P$. The set of linear extensions is called a realizer of the partially ordered set (X, P) .

Theorem 1 assures the existence of the dimension of any partial order.

4 Equivalence of Sequence Pair to Dimension of Partial Orders

Let X be a set of rectangles on a plane. Horizontal and vertical relations between the rectangles are defined as in Section 2. We consider two partial orders $P_x \subset X \times X$ and $P_y \subset X \times X$, which correspond to horizontal and vertical relations, respectively. We use $(i, j) \in P_x$ ($(i, j) \in P_y$) and $i <_x j$ ($i <_y j$) interchangeably unless $i = j$. The relation $i <_x j$ ($i <_y j$) implies rectangle j is right on (above) rectangle i .

Let P_1 and P_2 be two sequences of all the rectangles in X . As in (4) of Theorem 2, we can consider P_1 and P_2 as two linear orders (chains). For example, P_1 includes a set of rectangle pairs (i, j) such that $P_1 = (\dots, i, \dots, j, \dots)$ and rectangle pairs (i, i) for all rectangles i in X .

The reverse-ordered sequence of P_1 also defines a chain, and we denote it by P'_1 . On the basis of this notation, we can represent equations (1) and (2) in Section 2 as

$$P_x = P_1 \cap P_2, \quad (3)$$

$$P_y = P'_1 \cap P_2. \quad (4)$$

In equations (3) and (4) above, $\cap$ means set intersection. It should be noted that equations (3) and (4) mean that partial orders P_x and P_y have dimension two (See Definition 1).

The next theorem characterizes partially ordered sets of dimension two. Although the theorem is known, we give its proof to make the mathematical argument self-contained.

Theorem 3 [5] *Let G be the comparability graph of a poset (X, P) . Then, $\dim(X, P) \leq 2$ if and only if the complementary graph G^c is transitively orientable.*

Proof: Assume that $\dim(X, P) \leq 2$. If $\dim(X, P) = 1$, then P itself is a chain. The edge set of G^c is empty. If $\dim(X, P) = 2$, then there exists two linear extensions $\{L_1, L_2\}$ such that $L_1 \cap L_2 = P$. This implies $L_1 - P = (L_2 - P)^{-1}$. Here, $H^{-1} = \{(i, j) \mid (j, i) \in H\}$. Clearly, $L_1 - P$ can be considered an orientation of the complementary graph G^c . If $(i, j), (j, k) \in L_1 - P$ such that $i \neq j$ and $j \neq k$, then $(i, k) \in L_1$ because L_1 is a chain. If we assume that $(i, k) \in P$, then $(i, k) \notin L_1 - P$. This implies $(k, i) \notin L_2 - P$. Since $(j, i), (k, j) \in L_2 - P$, $(k, i) \in L_2$. Hence, $(k, i) \in P$ because if $(k, i) \notin P$, then $(k, i) \in L_2 - P$ and consequently $(i, k) \in L_1 - P$, which contradicts $(i, k) \notin L_1 - P$. Combining $(i, k) \in P$ and $(k, i) \in P$, we obtain $i = k$. This leads to $j = k$, which is a contradiction. Thus, $(i, k) \notin P$ holds, which leads to $(i, k) \in L_1 - P$. This implies that transitivity holds on $L_1 - P$. Hence, $L_1 - P$ is a transitive orientation of G^c .

Conversely, let F be a transitive orientation of G^c . A set of linear extensions $\mathcal{C} = \{P \cup F, P \cup F^{-1}\}$ satisfies $\cap \mathcal{C} = P$. Consequently, $\dim(X, P) = 2$. ■

As defined in [6], in the rectangle packing problem, given a set X of rectangles on a plane, we pack all of the rectangles into as small an enclosing rectangular area as possible without overlapping. It is clear that we can place all of the rectangles in X without overlapping if and only if every pair of rectangles has a horizontal or vertical relation as defined in Section 2. Furthermore, as shown in [6], it is sufficient to search for placements where any two rectangles have only a horizontal or only a vertical relation. The next theorem, which is obtained from Theorem 3, clarifies the mathematical essence of the sequence pair for rectangle packing.

Theorem 4 *Let X be a given set of rectangles in the rectangle packing problem. Three sets*

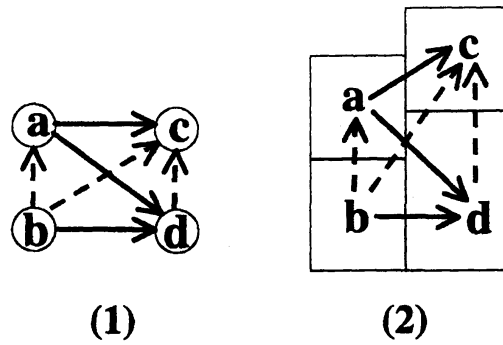


Figure 4: Partially ordered set of dimension two for horizontal relation (solid arrow) (1) Graph representation. (2) Rectangle packing.

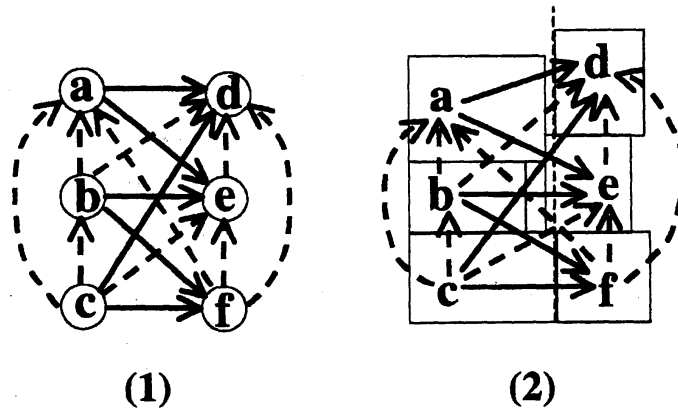


Figure 5: Partially ordered set of dimension three for horizontal relation (solid arrow). (1) Graph representation. (2) Rectangle packing.

of instances for rectangle packing satisfying (1), (2) and (3) below all coincide and include the optimal solution to the rectangle packing problem.

- (1) Partially ordered set P_x (P_y) of horizontal (vertical) relation has dimension two.
- (2) Every pair of rectangles i, j in X only has either a horizontal or a vertical relation.
- (3) For a sequence pair P_1 and P_2 , every pair of rectangles i, j has a horizontal or a vertical relation as defined by equation (1) or (2) in Section 2.

Figure 4 shows an instance of the rectangle packing problem. The horizontal (vertical) relation is depicted with solid (broken) arrows in Fig. 4. Let P_x (P_y) be a poset for the horizontal (vertical) relation. The poset P_x has dimension two, and its realizer is $P_1 = (a, c, b, d)$ and $P_2 = (b, a, d, c)$. Of course, this is an instance that must be searched in the rectangle packing problem. On the other hand, Fig. 5 demonstrates another instance in the rectangle packing problem for

six rectangles. The same symbols as in Fig. 4 are used in Fig. 5. The poset P_x for the horizontal relation has dimension three, and its realizer is $P_1 = (a, b, e, c, d, f)$, $P_2 = (b, c, f, a, d, e)$ and $P_3 = (c, a, d, b, e, f)$. If we change horizontal relation (c, d) to the vertical one, the dimension of P_x decreases to two. This corresponds to packing rectangle d in the horizontal direction so that it touches rectangle a . It is sufficient to examine only the resultant poset P_x of dimension two.

5 Applications to Rectangle Packing

A simulated annealing algorithm was applied to the rectangle packing problem. The sequence pair was used to construct the solution space for the simulated annealing algorithm. Figure 6 shows the simulated annealing algorithm. The number with parenthesis at the head of each line is only for reference. The outer **while** loop between lines (4) and (14) is repeated until a stopping criterion is satisfied. At each repetition, the temperature is lowered in accordance with a cooling schedule. On the other hand, in the inner **while** loop between lines (6) and (12), the solution is perturbed at random until the solution reaches some equilibrium at each temperature. The function $C(S)$ of solution S is a cost function, which calculates the area of the rectangle packing created by the solution S . As shown in line (9) to (11), the new solution S' is accepted with probability 1 if $\Delta \leq 0$, and with probability $e^{-\frac{\Delta}{T}}$ if $\Delta > 0$. The function $random(0, 1)$ generates a random number between 0 and 1. This procedure allows occasional "uphill moves", which worsen the current solution. The movement prevents the solution from being stuck at a locally optimal solution.

```

Simulated Annealing Algorithm
(1) begin
(2)    $S :=$  Initial solution  $S_0$ ;
(3)    $T :=$  Initial temperature  $T_0$ ;
(4)   while (stopping criterion is not satisfied) do
(5)     begin
(6)       while (not yet in equilibrium) do
(7)         begin
(8)            $S' :=$  Some random neighboring solution of  $S$ ;
(9)            $\Delta := C(S') - C(S)$  ;
(10)           $Prob := \min(1, e^{-\frac{\Delta}{T}})$ ;
(11)          if  $random(0, 1) \leq Prob$  then  $S := S'$ ;
(12)        end;
(13)      Update  $T$ ;
(14)    end;
(15)  Output best solution;
(16) end;

```

Figure 6: Simulated annealing algorithm

We can represent the solution by the sequence pair P_1 and P_2 , which are sequences of rectangle names. If necessary, as described in Section 2, the sequence pair can be easily transformed into the corresponding rectangle packing. The initial sequence pair made at line (2) in Fig. 6 is the one such that $P_1 = P_2$, which corresponds to a linear horizontal arrangement of rectangles. The random perturbations used at line (8) consist of three kinds of pair-interchanges: i) two rectangles in P_1 , ii) two rectangles both in P_1 and P_2 , and iii) the width and the height of rectangle, which optimizes the orientation of the rectangle. The temperature T was decreased exponentially. From a heuristic point of view, we made the probability of selecting the operation i) high in higher temperature, while the probability of selecting the operation iii) higher in lower temperature.

The algorithm was applied to the test data where 146 rectangles must be packed. The processing time on Sun Sparc-Station II was 29.9 minutes. The algorithm searched at most 606 192 distinct sequence pairs within the solution space whose size can be estimated by $(146!)^2 2^{146} \approx 1.23 \times 10^{552}$. It should be remarked that the search of only a fraction about 4.92×10^{-547} of the solution space was enough to reach the good packing result. For a larger test data where 500 rectangles must be packed, 18.83 hours were required with the same workstation. See [1] and [6] for details including the several figures of the rectangle packing results.

6 Conclusions

We have presented the mathematical background to an approach using the sequence pair for solving the rectangle packing problem. The equivalence of the sequence pair to the realizer of partial orders having dimension two was proved in connection with the algorithms for generating linear extensions of partial orders. We can say that the sequence pair for rectangle packing rediscovered a characterization of partial orders of dimension two. The sequence pair provides an efficient representation of instances of the rectangle packing problem, but it also gives a compact data structure that can be extended to solve more complex problems. On the basis of the mathematical background described in this paper, we intend to study its extensions to various problems such as three-dimensional packing and its analysis methods.

References

- [1] H. Murata, K. Fujiyoshi, S. Nakatake and Y. Kajitani, "VLSI module placement based on rectangle-packing by the sequence-pair," *IEEE Trans. on Computer-Aided Design of*

- Integrated Circuits and Systems*, vol. 15, no. 12, pp. 1518–1524, Dec. 1996.
- [2] Y.Z. Liao and C.K. Wong, "An algorithm to compact a VLSI symbolic layout with mixed constraints," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, vol. 2, no. 2, pp. 62–69, Apr. 1983.
 - [3] A. V. Aho, J. E. Hopcroft and J. D. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley Publishing Company, Massachusetts, 1974.
 - [4] M. C. Golumbic, *Algorithmic Graph Theory and Perfect Graphs*, New York, Academic Press, 1980.
 - [5] B. Dushnik and E. W. Miller, "Partially ordered sets," *American Journal of Mathematics*, vol. 63, pp. 600–610, 1941.
 - [6] Y. Kajitani, "On packing of rectangles into a small plane area," *IPSJ SIG Notes*, DA-89-2, pp. 7–14, Sept. 1998.