Title	Optimal Space Partitioning Method Based on Rectangular Duals of Planar Graphs
Author(s)	Fujita, Kikuo; Akagi, Shinsuke; Shimazaki, Sadao
Citation	JSME international journal. Ser. C, Dynamics, control, robotics, design and manufacturing. 1996, 39(2), p. 378-386
Version Type	VoR
URL	https://hdl.handle.net/11094/2954
rights	
Note	

Optimal Space Partitioning Method Based on Rectangular Duals of Planar Graphs*

Kikuo FUJITA**, Shinsuke AKAGI**
and Sadao SHIMAZAKI***

An optimal space partitioning method is proposed based on rectangular duals of planar graphs and a simulated annealing algorithm. The layout problem, in which a region should be partitioned into plural subregions of layout components so as to satisfy relationships between neighbors and size conditions for a whole region and respective subregions, occurs in several layout designs. Such layout problems are characterized by the combinatorial property on topological structure among subregions. In our method, rectangular duals of planar graphs are used for representing such a structure, and it is optimized through a simulated annealing algorithm. In the annealing process, topological layouts represented by rectangular duals of planar graphs are manipulated with rules, each of them is embodied into an actual layout using the generalized reduced gradient method which is one of the numerical optimization techniques for constrained nonlinear optimization problems. Finally, we show an example of an access control room layout in a power plant design in order to check the effectiveness and validity of the proposed method.

Key Words: Optimal Space Partition, Rectangular Dual, Simulated Annealing, Optimization, Design Engineering, Layout Design

1. Introduction

In several kinds of layout problems, a region must be partitioned into plural subregions of layout components so as to satisfy relationships between neighbors and size conditions for a whole region and respective subregions. Such problems occur in several plant layout design and architectural layout problems. However, it is difficult to find optimal layouts for these problems, because it is necessary to satisfy the above layout conditions in a compromised way and it includes a number of combinatorial and topological conditions concerning the location of subregions within a layout.

In this paper, we propose a layout approach for such space partitioning problems by introducing the method of rectangular duals of planar graphs[1],[2] and its extension by Tamiya[3] for representing topological neighboring relationships, a simulated annealing algorithm (SA)[4],[5] for optimizing such a neighboring structure, and the generalized reduced gradient (GRG) method[6], a numerical optimization algorithm for nonlinear constrained optimization problems, for generating an embodiment layout from such a structure. Moreover, we apply it to a space partitioning problem for equipment in a power plant in order to ascertain its validity and effectiveness.

A layout can be hierarchically represented by combining topological relationships among layout components and their exact positions and directions. And a hybrid approach of topological search methods and numerical sizing optimization methods is used in the layout for a power plant design problem with a constraint-directed search technique and an optimization procedure[7],[8], and is the optimal nesting method with a genetic algorithm and a local optimization algorithm[9].

* Received 14th April, 1995. Japanese original: Trans. Jpn. Soc. Mech. Eng., Vol. 60, No. 579, C (1994), pp. 3662-3669. (Received 17th February, 1994)

** Department of Mechanical Engineering for Industrial Machinery & Systems, Faculty of Engineering, Osaka University, 2-1 Yamadaoka, Suita, Osaka 565, Japan
*** Graduate School, Osaka University

2. Space Partitioning Problems and Fundamental Methods

2.1 Space partitioning problems

As mentioned in the Introduction, the space partitioning problem discussed in this paper is a layout design problem in which a certain region is partitioned into plural rectangular subregions. Its characteristics are summarized as follows.

- Form and dimension of a whole region are fixed as given conditions.
- While each subregion is constrained with minimum space area, its dimensions are free, unless it becomes too long, that is, unless the aspect ratio becomes extreme.
- Several pairs of subregions should be directly connected with each other, corresponding to activities within a whole region. In the following, such neighboring relationships are called "*neighboring conditions*" in order to distinguish them from the situation that a pair is accidentally neighbored.

This layout problem is a complicated problem which includes many combinatorial conditions concerning topological structure, which is similar to other kinds of layout problems. In addition to such difficulties, this problem is more difficult, because the dimensions of respective subregions are freer, than those in layout problems in VLSI design.

In this paper, we propose an approach for searching for a quasi-optimal solution of the above space partitioning problem by integrating the following methods: the representation of combinatorial and topological structure of layouts using rectangular duals of planar graphs[1],[2], a simulated annealing (SA) algorithm[4],[5] for optimizing such a combinatorial aspect of layouts, and the generalized reduced gradient (GRG) method[6] which is a nonlinear constrained optimization algorithm for dealing with the conditions related to aforementioned space areas.

2.2 Simulated annealing

Simulated annealing[4],[5] is an optimization algorithm which is analogous to the simulation of the annealing of solids. It can be used to search for a quasi-optimal solution in combinatorial and multipeak optimization problems, which are difficult using traditional hill-climbing algorithms. In the optimization process of the annealing algorithm, first a solution is assumed, and then another tentative solution is generated in its neighborhood. If the cost function is improved with the tentative solution, it is adopted as a solution. Even if the cost function is not improved, the tentative solution is also adopted with a probability. This probability is gradually decreased during the optimization process, which is controlled by a parameter called the *temperature* as an analog to the physical annealing process. Based on this control mechanism, first solutions are globally searched, and then the search region is gradually limited to a local space. This mechanism is expected to result in a globally optimal solution.

Simulated annealing has been applied to the traveling salesman problem[4] and cell layout problems in VLSI design[3],[4],[10]. In the field of mechanical design, it is also applied to, for example, the optimization problem of gear teeth numbers of a multispeed gearbox[11], nesting problems[12] and 3-dimensional component layout problems[13],[14]. These applications show the effectiveness of the algorithm for solving various combinatorial optimization problems. In order to effectively apply it to individual problems, it is necessary to adjust the acceptance probability of new solutions, e.g., *temperature*, during the optimization process. Suitable representation of an optimization problem is also required, and a neighborhood structure of solutions must be imposed on such a representation so as to correspond well with underlying properties of individual problems. Therefore, such a representation method should be suitably introduced for application to the space partitioning problem.

2.3 Graph-based representation of layouts

The method of rectangular duals of planar graphs is proposed in order to efficiently manipulate neighboring relationships on the layout problems composed of rectangular regions, in which such relationships are represented with a 4-connected triangulated plane graph. That is, topological structure of a layout is represented with such a graph, and the manipulation of layouts can be done through the manipulation of graphs, since an actual layout and a corresponding topological graph can be translated to each other. Kozminski et al.[1],[2] showed the characteristics of such graphs and an algorithm for space partitioning. Tamiya[3] proposed the transformation method of such graphs for solving macrocell layout problems using a simulated annealing algorithm by extending such dual graphs to directional graphs.

The graph representation method also is considered to be effective for the space partitioning problems, the properties of which were shown in section 2.1, since the neighboring relationships of a layout can be immediately transformed through the graph. However, some extensions and modifications are required before it can be applied to the space partitioning problem, because in the case of cell layout problems, the dimensions of respective subregions are strictly predefined and the fixed neighboring relationships are not constrained. In this paper, we extend the representation so as to overcome these differences in order to apply it to the space partitioning problem, which will be shown in the following sections.

3. Rectangular Duals of Planar Graphs and Representation of Layouts

3.1 Layout representation by rectangular duals of planar graphs

Figure 1 shows how to represent a layout in the space partitioning problem with rectangular duals of planar graphs[3], where (a) shows the graph representing topological structure of a layout, and (b) its corresponding actual layout. In (a), the numbered nodes, ①, ②, ③ etc., indicate subregions to be located, and Ⓔ, Ⓦ, Ⓢ and Ⓝ correspond to boundaries in the east, west, south and north sides of the whole region. Neighboring relationships among those nodes are represented with two kinds of edges among them. A horizontal edge, '⟶', means that a pair of subregions connected with it are touching each other from west to east. A vertical edge, '- - -→', means that a pair of subregions connected with it are touching each other from south to north.

The characteristics of the rectangular duals of planar graphs are summarized as follows[3].

- There is no intersection among subregions, and there is no empty space among them.
- The shape of the whole region is rectangular.
- A pair of nodes connected with an edge in a graph representation are guaranteed to touch each other.

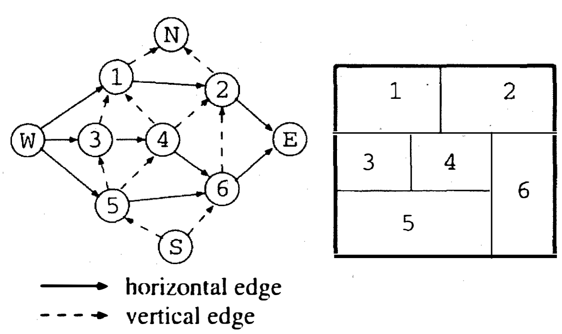
Because of these characteristics, the layout representation method with the rectangular duals of planar graphs is considered to be very effective for the space partitioning problem based on its characteristics described in section 2.1.

3.2 Transformation of topological structure of layouts

In order to optimize the above graph structure using a simulated annealing algorithm, it is required to transform a layout into another one in its neighbors via some rules.

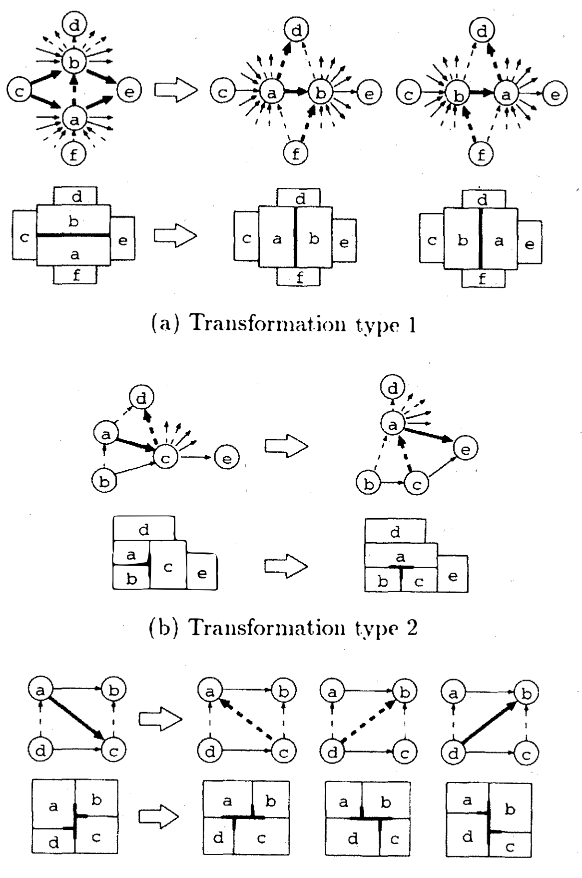
Figure 2 shows such transformation rules introduced by Tamiya[3], in which neighboring relationships among several subregions are locally changed. In the figure, *transformation type 1* transforms a horizontal (vertical) relationship between a pair of subregions into a vertical (horizontal) relationship by orthogonally rotating the topological structure. *Transformation type 2* transforms the direction of T-type neighboring relationships among three subregions at their corners. *Transformation type 3* transforms the direction of neighboring relationships among four subregions at their corners. In addition, there are two, eight and four similar sets of rules for the respective types.

In the process of the simulated annealing algorithm, neighborhood layout solutions are generated according to the transformation rules shown in Fig. 2. That is, subgraphs matched with preceded patterns shown in the left sides of the figure are first listed, and



(a) Graph (b) Layout

Fig. 1 Example of rectangular duals of planar graphs



(a) Transformation type 1

(b) Transformation type 2

(c) Transformation type 3

Fig. 2 Transformations of rectangular duals of planar graphs then such subparts of a graph are individually transformed into the neighboring layouts according to the corresponding transformation rules. The optimal neighboring structure among subregions may be determined by means of a simulated annealing algorithm with these transformations, since it was proved that all rectangular duals of planar graphs with a certain number of nodes may be transformed into each other by a finite number of transformation operations[3].

3.3 Embodying layouts by the generalized reduced gradient method

An actual layout, which is represented with positions and dimensions of respective subregions, is determined from the topological structure represented using rectangular duals of planar graphs by optimizing a constrained nonlinear minimization problem with the generalized reduced gradient method (GRG)[6]. The formulation of this optimization problem is shown in the following.

3.3.1 Design variables As shown in Fig. 3, four design variables, x_i^+, x_i^-, y_i^+ and y_i^- are defined for the position and dimension of each subregion, Region$_i$ ($i=1, \cdots, N$), under the orthogonal coordinate system.

3.3.2 Constraints The following two types of constraints are considered.

• Pairs of neighboring subregions in the rectangular duals of a planar graph should also be neighbored in an actual layout and the length of the shared edge should be more than the assigned length. For example, in the case that Region$_j$ is the east-side neighbor of Region$_i$ as shown in Fig. 4, the following equations should be satisfied.

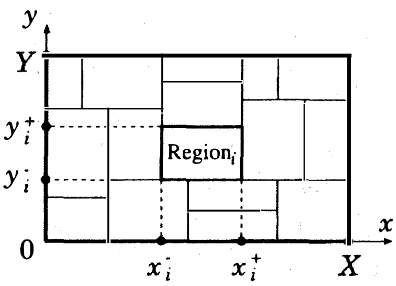


Fig. 3 Coordinate system and design variables

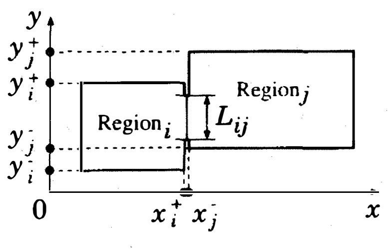


Fig. 4 Constraint between neighboring pair of subregions

$$\left.\begin{aligned} &x_i^+ = x_j^- \\ &y_i^+ - y_j^- \geq L_{ij} \\ &y_j^+ - y_i^- \geq L_{ij} \end{aligned}\right\} \qquad (1)$$

Where, L_{ij} is the minimum length of the shared edge between Region$_i$ and Region$_j$.

If any activity exists between the subregions, L_{ij} is set to the width of the pathway. Otherwise, L_{ij} is set to 0.

• In order to maintain the minimum space area required for each subregion, for example, the following equations are introduced for Region$_i$.

$$\left.\begin{aligned} &(x_i^+ - x_i^-)(y_i^+ - y_i^-) \geq S_i \\ &x_i^+ - x_i^- \geq L_i \\ &y_i^+ - y_i^- \geq L_i \end{aligned}\right\} \qquad (2)$$

Where, S_i is the minimum required space area of Region$_i$, and L_i is the minimum length of each edge of Region$_i$.

3.3.3 Objective function The objective function is defined as follows with the position variables, X and Y, which are shown in Fig. 3.

$$(X - X_S)^2 + (Y - Y_S)^2 \rightarrow Min. \qquad (3)$$

Where, $X_S \times Y_S$, is the dimension of the whole region where subregions should be arranged. This is given as one of the layout conditions.

The formulation shown here is a constrained nonlinear minimization problem. By solving this problem using the generalized reduced gradient method[6], an actual layout, as shown in Fig. 1(b), can be determined from the topological layout shown in Fig. 1(a).

4. Optimal Space Partitioning Method

In this section, we show the validity and effectiveness of the optimal space partitioning method by using a simulated annealing algorithm based on the representation method with the rectangular duals of planar graphs and embodiment procedure with the generalized reduced gradient method.

4.1 Outline of optimal space partitioning method

The optimal space partitioning method consists of the following two steps.

(I) A feasible graph which satisfies the neighboring conditions is searched.

(II) The simulated annealing algorithm is applied to such an initial layout so as to search for a quasi-optimal layout.

The reason that a feasible graph is searched first is as follows. In the space partitioning problem, it is required to satisfy the neighboring conditions among certain pairs of subregions. The constraints and objective functions mentioned in the previous section make no sense for a layout which violates such neighboring conditions. In order to avoid this situation, the transformations of layouts during the simulated annealing process should be limited among feasible layouts. Therefore, a feasible layout is required as the initial layout for the simulated annealing algorithm.

The details of the above two steps are shown in the following subsections.

4.2 Generation of initial feasible layout

First, an initial feasible graph is searched by a simple hill-climbing algorithm as follows, where the number of unsatisfied neighboring conditions is taken as the objective function to minimize.

First, an initial graph is randomly generated. Then, trial graphs are generated by iteratively applying the transformation rules of graphs shown in Fig. 2. If the number of unsatisfied neighboring conditions is reduced by the trial, then it is acceptable as a graph. Otherwise, it is not. When the graph which satisfies all of the conditions is found during the iterations of these operations, a feasible layout is generated.

In order to search for a feasible graph with fewer trials, the subregions are divided into blocks based on the neighboring conditions, and the above hill-climbing algorithm is hierarchically applied to both blocks and graphs of respective sets of subregions within individual blocks.

(1) The blocks are defined by recognizing a set of subregions as a block by merging leaf nodes in a graph of neighboring conditions into trunk nodes and by unifying trunk nodes which are directly connected in series into a node. The graph among blocks is also defined through this procedure. Figure 5 shows an example block graph of the application which will be explained in section 5.

(2) The neighboring conditions among blocks are defined based on the original neighboring conditions among subregions. A feasible rectangular dual graph among blocks which satisfy such conditions is searched using the simple hill-climbing algorithm.

(3) The respective rectangular dual graphs among subregions within individual blocks are searched using the simple hill-climbing algorithm. When searching, four boundary nodes are assumed, based on the neighboring relationships with other blocks. As a result, the generated feasible subgraphs are expected to satisfy both original neighboring conditions within blocks and neighboring conditions at the boundary assigned from neighboring relationships among blocks.

(4) The graph among blocks and the subgraphs within them are merged into a whole graph. If it satisfies all of the original neighboring conditions, it can be used as an initial feasible layout of the simulated annealing algorithm, and the procedure is stopped. Otherwise, the procedure goes to Step (5).

(5) The transformation rules shown in Fig. 2 are randomly applied to the graph among blocks in order to find another feasible graph. Then, the procedure goes to Step (3).

Among these operations, the operation of Step (4), which merges subgraphs into a whole graph, is done by iterating the operations, as shown in Fig. 6. That is, in the case that Block A and Block B are connected from west to east, as shown in (a), and the subgraphs within respective blocks are fixed, as shown in (b), the edges between nodes a_1 and a_2 in Block A and nodes b_1 and b_2 in Block B are assumed as their candidates as shown in (c). Among these candidates, those with conflicted pairs of edges are neglected so as to satisfy the planarity of the graph and the original neighboring conditions. By means of

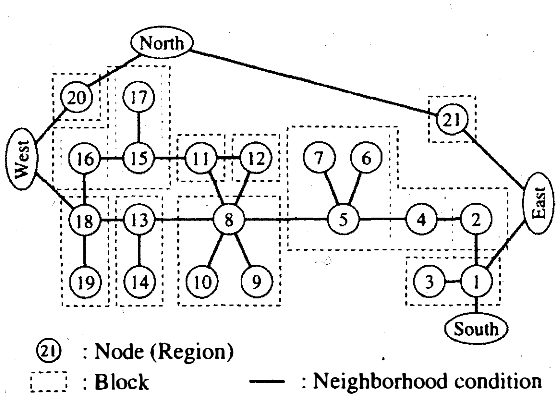
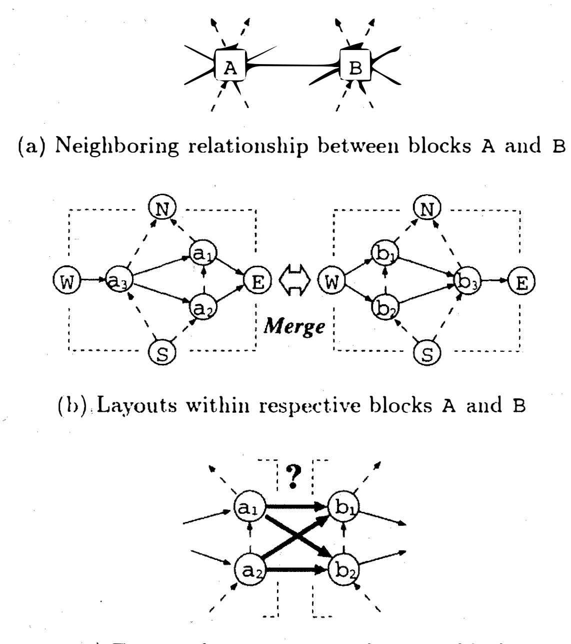


Fig. 5 Example of blocks in layout structure



(a) Neighboring relationship between blocks A and B

(b) Layouts within respective blocks A and B

(c) Fixing edges connecting between blocks

Fig. 6 Merging method of block layouts

these operations, the subgraphs which are generated by the operations of Steps (2) and (3) are merged into a whole rectangular dual graph among subregions.

4.3 Optimization of topological structure through a simulated annealing algorithm

The optimal space partition is searched by the simulated annealing algorithm[4],[5] from a feasible initial graph generated by the procedures shown in the above subsection. This search procedure is shown in Fig. 7, which is as follows.

(1) One of feasible layouts which satisfy the required neighboring conditions is generated by the simple hill-climbing algorithm above. The cost of such a layout is calculated with Eq. (4), mentioned later, and its value is represented by C.

(2) The initial temperature of the simulated annealing algorithm is set to T.

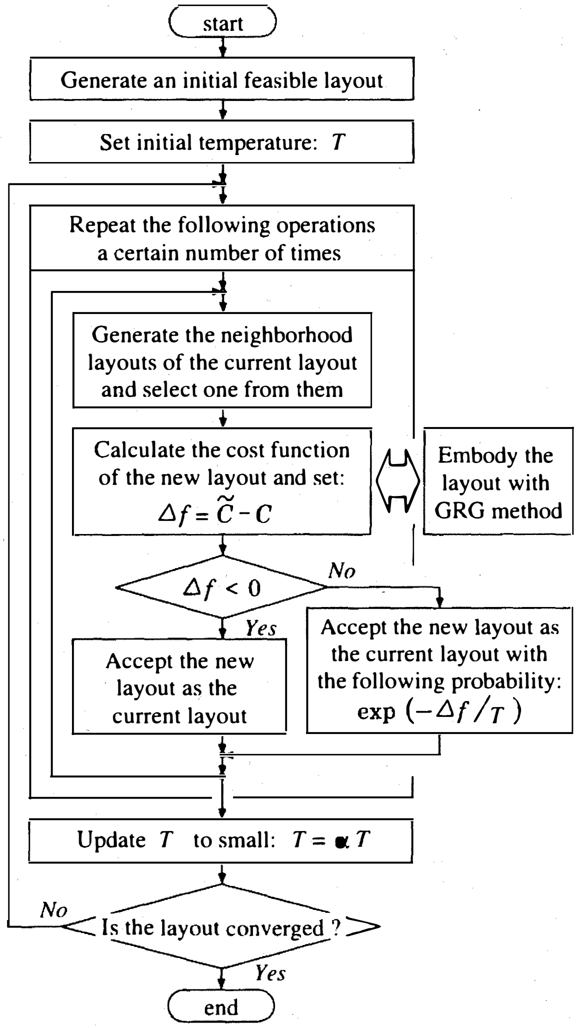


Fig. 7 Optimization of topological structure through simulated annealing algorithm

(3) A primary candidate list of layouts is generated by applying the transformation rules shown in section 3.2, and those which do not satisfy the neighboring conditions are eliminated from the candidate list. A tentative layout is randomly selected from such candidates, and the actual layout corresponding to it is fixed by the embodiment procedure shown in section 3.3. The cost of such an actual layout is calculated with Eq. (4), and its value is represented by $\tilde{C}$.

(4) Under $\Delta f = \tilde{C} - C$, if $\Delta f < 0$, then the tentative layout is accepted as a new layout, and $\tilde{C}$ is set to a new value of C. If $\Delta f \geq 0$, then it is accepted as the new layout within the probability $p = \exp\left(-\frac{\Delta f}{T}\right)$.

(5) After the operations from (3) to (4) have been iterated a certain number of times, the procedure goes to Step (6). Otherwise, the procedure returns to Step (3).

(6) The temperature T is decreased by multiplying it by the temperature update factor α.

(7) If the layout is recognized to have converged, then the procedure is terminated. Otherwise, the procedure returns to Step (3).

With these operations, a suitable layout is expected to be generated by iteratively improving layouts through annealing operations.

4.4 Evaluation of space partition

The following C is used here as a cost function, where the method of evaluating a layout with a cost function is dependent on individual applications.

$$C = w_1 \sum_{i=1}^{N} surplus_i + w_2 \sum_{i=1}^{N} ratio_i \tag{4}$$

Where, $surplus_i$ and $ratio_i$ are the objective functions concerning surplus area and aspect ration of Region$_i$, respectively. w_1 and w_2 (w_1, $w_2 > 0$) are the weighting factors on these terms, respectively.

Surplus area here means excess space area as compared with the required minimum space area. This objective function is calculated for a subregion, Region$_i$, with the following equation under actual space area S_i.

$$surplus_i = \begin{cases} S_i - \hat{S}_i \cdots S_i > \hat{S}_i \\ 0 \quad\quad \cdots S_i \leq \hat{S}_i \end{cases} \tag{5}$$

Where, $\hat{S}_i$ is the required minimum space area given as a specification.

The objective function concerning the aspect ratio, $R_i (\geq 1)$, is defined by the following function.

$$ratio_i = \begin{cases} (R_i - \hat{R}_i)^2 \cdots R_i > \hat{R}_i \\ 0 \quad\quad \cdots R_i \leq \hat{R}_i \end{cases} \tag{6}$$

$$\hat{R}_i = R_{min} \times \left(\frac{\hat{S}_i}{S_{min}}\right)^{\frac{1}{\beta}} \tag{7}$$

Where, R_{min} is the acceptable maximum value of the aspect ratio of the subregion, the required space area of which is minimum, and S_{min} is the minimum value of required space areas. β $(1 \leq \beta \leq \infty)$ is a coefficient used for setting individual acceptable maximum values of the aspect ratio for respective subregions. While the effect of β is practically dependent on the values of $\hat{S}_i$, if β is nearly equal to 1, the equation does not yield the aspect ratio itself, and lengths of shorter edges of all subregions are made to be greater than a common value. On the other hand, if $\beta = \infty$, all $\hat{R}_i$ are equal to each other, which means that the aspect ratio is similarly evaluated among every subregion. Consequently, by taking a suitable value between 1 and ∞, a layout with a good balance is expected, where subregions with less space area tend to be almost square and ones with more space area tend to be elongated.

4.5 Implementation of the space partitioning method

Computer implementation of the space partitioning method is shown here. The proposed method is composed of two parts: optimization of topological structure through simulated annealing operations and embodiment of actual layouts from such graphs. The former is characterized by transformation operations of graphs, for which symbol manipulation functions in Lisp are suitable. The latter is characterized by numerical calculation for optimization, which can be efficiently implemented in C programming language. The whole method is implemented by communicating between these two procedures through the socket library[15] of the UNIX operating system. The actual implementation of the method and its execution are done on a UNIX workstation, the Sun SPARC Station (SunOS 4.1.1).

5. Example of Space Partitioning Results

Finally, the space partitioning method is applied to the layout problem of an access control room in a power plant. In this kind of layout problem, it is required to satisfy several neighboring conditions between subregions which correspond to certain activities and the space condition that each subregion has a certain minimum space area.

Twenty-one subregions to be arranged are listed in Table 1 with their space area conditions, $\hat{S}_i$, and aspect ratio conditions, $\hat{R}_i$. Figure 5 shows the neighboring conditions to be satisfied among such subregions. Figure 8 shows a layout result generated by the proposed method. This result is ascertained to satisfy the above neighboring conditions and to achieve a good balance concerning the aspect ratio of respective subregions. Figure 9 shows the convergence history of the cost function, C, given by Eq. (4), (a) against the acceptance times of a new layout solution, and (b) against the times of temperature

Table 1 Layout conditions

No.	Name of subregion	$\hat{S}_i$	$\hat{R}_i$
1	atlb-pre	49.00	2.74
2	atlb-pro	40.00	2.56
3	cont-man	13.00	1.76
4	shoes-box	65.00	3.02
5	locker-r	204.00	4.42
6	lavatory	16.00	1.89
7	relax-r	36.00	2.48
8	watchman-r	68.00	3.06
9	watchman-wait-r	8.00	1.50
10	special-wait-r	14.00	1.81
11	monitor-pre	58.00	2.90
12	monitor-pro	58.00	2.90
13	clothing-r	89.00	3.35
14	clothing-store	105.00	3.54
15	hand-washing-r	71.00	3.11
16	dressing-r	84.00	3.28
17	shower-r	37.00	2.50
18	yellow-shoes	92.00	3.39
19	patrol-r	29.00	2.30
20	storehouse	15.00	1.85
21	empty-zone	62.00	2.97

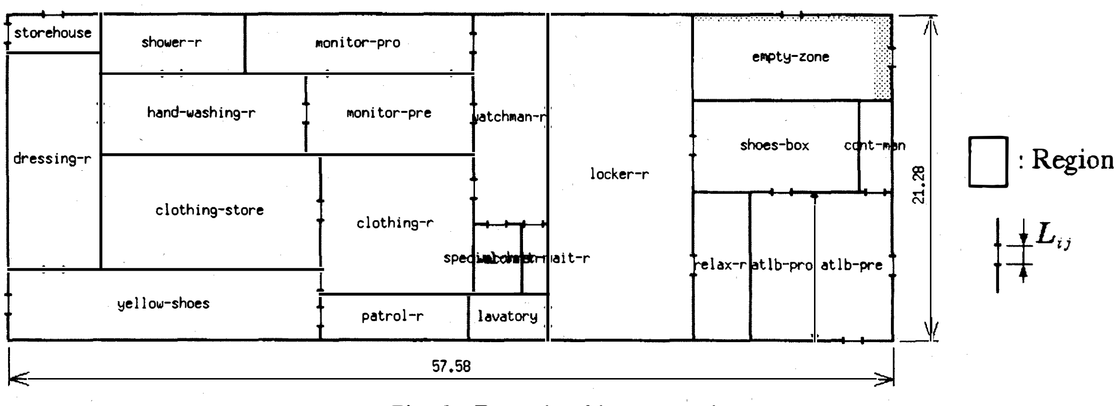
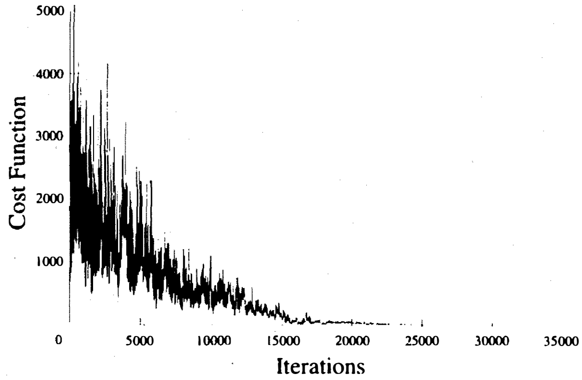
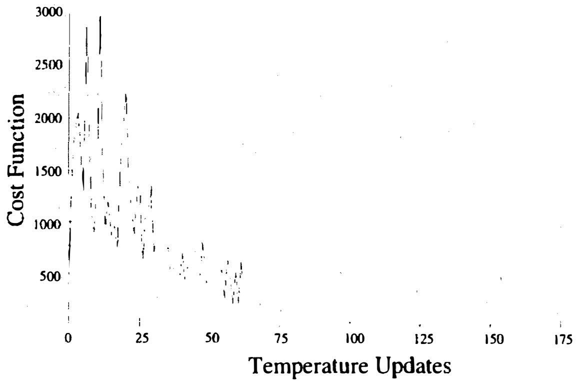


Fig. 8 Example of layout results



(a) Objectives versus acceptances of new layouts



(b) Objectives versus respective temperatures

Fig. 9 Convergence of simulated annealing algorithm

updates. This figure demonstrates how the layout solutions gradually converge to an optimal one.

This layout example is generated under the following conditions: initial temperature $T=1000$, iteration times at one temperature, 200; temperature update factor $\alpha=0.95$; weighting factors $w_1=1.0$ and $w_2=2.0$; and the parameters for evaluating aspect ratio $R_{min}=1.5$ and $\beta=3$.

6. Conclusion

In this paper, we proposed an optimal space partitioning method in which topological structure of layouts is represented by rectangular duals of planar graphs, and such structure is optimized by a simulated annealing algorithm. Moreover, its validity is ascertained by applying it to the layout problem of an access control room in a power plant.

The representation with rectangular duals of planar graphs is fairly effective for manipulating neighboring structures in space layout problems, because it can distinguish topological layouts from actual layouts. Simulated annealing is also effective for optimizing such topological structure. Additionally, it is ascertained that the cost function, which is derived from terms related to both spare space area and aspect ratio, is effective in generating well-balanced layouts.

We thank Tetsundo Nakatogawa, Kohsuke Yasuda and Kaname Shibato of Mitsubishi Atomic Power Plant Industries Inc. for their helpful instruction on the space partitioning problem shown in section 5.

References

(1) Kozminski, K. and Kinnen, E., An algorithm for finding a rectangular dual of a planar graph for use in area planning for VLSI integrated circuits, Proceedings of the 21st Design Automation Conference, IEEE and ACM, (1984), p. 655.

(2) Kozminski, K. and Kinnen, E., Rectangular Duals of Planar Graphs, Networks, Vol. 15 (1985), p. 145.

(3) Tamiya, Y., Placement based on Rectangular Dual, Research Reports of Information Processing Society of Japan, (In Japanese), Vol. 91, No. 35 (DA-57), (1991), p. 57 3 1.

(4) Kirkpatrick, S., Gelatt Jr., C.D. and Vecchi, M.P., Optimization by Simulated Annealing, Science, Vol. 220, No. 4598, (1983), p. 671.

(5) van Laarhoven, P.J.M. and Aarts, E.H.L., Simulated Annealing: Theory and Applications, (1987), D. Reidel Publishing.

(6) Gabriele, G.A. and Ragsdell, K.M., The Generalized Reduced Gradient Method: A Reliable Tool for Optimal Design, Transactions of the ASME, Journal of Engineering Industry, Vol. 99, (1977), p. 394.

(7) Akagi, S. and Fujita, K., A Layout Design System based on Constraint Directed Reasoning (1st Report: The Fundamental Approach), Trans. Jpn. Soc. Mech. Eng., (In Japanese), Vol. 56, No. 528, C (1990), p. 2286.

(8) Fujita, K., Akagi, S., Hase, H., Nakatogawa, T. and Takeuchi, M., Plant Layout Design System using a Hybrid Approach with a Constraint Directed Search and a Mathematical Optimization Technique, JSME Int. J., Ser. C, Vol. 36, No. 4 (1993), p. 562.

(9) Fujita, K., Akagi, S. and Hirokawa, N., Approach for Optimal Nesting Using Genetic Algorithm and Local Minimization Algorithm, Trans. Jpn. Soc. Mech. Eng., (In Japanese), Vol. 59, No. 564, C (1993), p. 2576.

(10) Sechen, C. and Sangiovanni Vincentelli, A., The Timber Wolf Placement and Routing Package, IEEE Journal of Solid-State Circuits, Vol. SC-20, No. 2, (1985), p. 510.

(11) Jain, P. and Agogino, A.M., Optimal Design of Mechanisms Using Simulated Annealing: Theory and Applications, Advances in Design Automation—1988—, DE-Vol. 14, (1988), ASME, p. 233.

(12) Jain, P., Fenyes, P. and Richter, R., Optimal Blank Nesting Using Simulated Annealing, Advances in

Design Automation—1990—, DE-Vol. 23-2 (1990), ASME, p. 109.
(13) Sugiura, H., Jang, S. and Nakajima, N., Application of the Simulated Annealing Method to the 3-Dimensional Layout Design Problem (Optimization of the 3-Dimensional Layout Design Problem of Submergible Boat), Trans. Jpn. Soc. Mech. Eng., (In Japanese), Vol. 58, No. 546, C (1992), p. 352.
(14) Szykman, S. and Cagan, J., Automated Generation of Optimally Directed Three Dimensional Component Layouts, Advances in Design Automation—1993—, DE-Vol. 65-1, (1993), ASME, p. 527.
(15) Curry, D.A., Using C on the Unix System—A Guide to System Programming. (1989), O'Reilly & Associates, Inc.