IEMS Vol. 8, No. 1, pp. 22-28, March 2009.

Efficient Algorithms for Solving Facility Layout Problem Using a New Neighborhood Generation Method Focusing on Adjacent Preference

Tatsuya Fukushi†
Graduate School of System Design, Tokyo Metropolitan University
Asahigaoka 6-6, Hino, Tokyo, Japan
Tel: (+)81-42-585-8606, E-mail: fukushi-tatsuya@sd.tmu.ac.jp

Hisashi Yamamoto
Graduate School of System Design, Tokyo Metropolitan University
Asahigaoka 6-6, Hino, Tokyo, Japan
Tel: (+)81-42-585-8606

Atsushi Suzuki
Faculty of Humanities, Hirosaki University
Bunkyocho 1, Hirosaki, Aomori, Japan

Yasuhiro Tsujimura
Faculty of Engineering, Nippon Institute of Technology
Gakuendai 4-1, Miyashiro, Saitama, Japan

Received January 22, 2008/Accepted September 14, 2008 (Selected from APIEMS 2007)

Abstract. We consider facility layout problems, where *mn* facility units are assigned into *mn* cells. These cells are arranged into a rectangular pattern with *m* rows and *n* columns. In order to solve this cell type facility layout problem, many approximation algorithms with improved local search methods were studied because it was quite difficult to find exact optimum of such problem in case of large size problem. In this paper, new algorithms based on Simulated Annealing (SA) method with two neighborhood generation methods are proposed. The new neighborhood generation method adopts the exchanging operation of facility units in accordance with adjacent preference. For evaluating the performance of the neighborhood generation method, three algorithms, previous SA algorithm with random 2-opt neighborhood generation method, the SA-based algorithm with the new neighborhood generation method (SA1) and the SA-based algorithm with probabilistic selection of random 2-opt and the new neighborhood generation method (SA2), are developed and compared by experiment of solving same example problem. In case of numeric examples with problem type 1 (the optimum layout is given), SA1 algorithm could find excellent layout than other algorithms. However, in case of problem type 2 (random-prepared and optimum-unknown problem), SA2 was excellent more than other algorithms.

Keywords: Facility Layout, Simulated Annealing, Neighborhood Generation Method

1. INTRODUCTION

Facility layout problems are classified into "equal-size facility type" and "unequal-size facility type." The size (area) of all facility units is equal in the problem of equal-size type, and, the size of facility unit is different in the case of unequal-size type. As other classification, there are "equal-shape type" and "unequal-shape type." In the case of equal size and equal shape type, the facility layout problem can be represented as quadratic assignment problem (QAP) (Koopmans and Beckman, 1957; Burkard, 1984; Golany and Rosenblatt, 1989; Lacksonen and Enscore Jr., 1993). The complexity of the approach is lower than unequal-size and unequal-

† : Corresponding Author

shape type. However, if the number of units becomes large, it is impossible to obtain the exact solution. Then, various approaches based on heuristic methods are proposed at present.

Generally, in the facility layout problem, minimizetion of total cost or total time of material flow is used in the objective function, it depends on distance between facility units. On the other hand, satisfaction or goodness of closeness between pairs of facility units is adopted as the criterion for evaluation of layout plans (Foulds, 1983; Houshayar and White, 1993). Using both, there are studies as multi criteria (or multi goal) facility layout problems (Rosenblatt, 1979; Dutta and Sahu, 1982; Fortenberry and Cox, 1985; Urban, 1987; Harmonosky and Tothero, 1992).

In this paper, we consider facility layout problems, where *mn* facility units (machines) are assigned into *mn* cells. These cells are arranged into a rectangular pattern with *m* rows and *n* columns. And, as the objective function, we consider material handling cost and adjacent factors between machines. In these facility layout problems, we aim to obtain the layout of machines, which has the minimum objective value, namely the optimal layout. To solve such problems, the algorithm based on genetic algorithm (GA) and tabu search (TS) had been proposed (Suzuki *et al*., 2005).

The purpose of this paper is to propose a new algorithm for obtaining pseudo-optimal solutions in the above-mentioned facility layout problem. Our algorithm is based on Simulated Annealing (SA) method (Kirkpatrick *et al*., 1983). We conducted a numerical experiment to evaluate the performance of the algorithm. As a result, we could get better solution than the existing method and find out our proposed algorithm is effective.

2. THE CELL TYPE FACILITY LAYOUT PROBL EM

The facility layout problem considered in this paper is represented as the 2-dimensional layout problem to assign the *mn* facility units into the *mn* cells (*m* rows and *n* columns) as shown in Figure 1, where each facility unit must be assigned into one cell and just one facility unit must be assigned into each cell (The "cell" doesn't mean the cell in cellular manufacturing systems). The cell on *i*-th row and *j*-th column is called as Cell (*i, j*). In this problem, the layout plans are evaluated by material handling cost and adjacent factor between facility units.

Before stating our considered facility layout problem, we define the following notations.

For $i = 1, 2, \cdots, m$ and $j = 1, 2, \cdots, n$

$p(i, j)$: Facility unit No. assigned to Cell(*i, j*), where *i* means row number of the cell and *j* column number

If facility unit *k* is assigned to Cell(*i, j*), then $p(i, j) = k$.

Let *P* be *mn* dimension vector, $(p(1, 1), p(1, 2), \cdots p(m, n))$.

For $i_1 = 1, 2, \cdots, m$, $i_2 = 1, 2, \cdots, m$, $j_1 = 1, 2, \cdots, n$, $j_2 = 1, 2, \cdots, n$,

$D((i_1, j_1), (i_2, j_2))$: distance between Cell (i_1, j_1) and Cell (i_2, j_2) .

For $k_1 = 1, 2, \cdots, m$, $k_2 = 1, 2, \cdots, m$

$F(k_1, k_2)$: material handling cost per unit distance from Facility unit k_1 to Facility unit k_2 .

$G(k_1, k_2)$: adjacent factor between Facility units k_1 and k_2 , if Facility units k_1 and k_2 are adjacent to each other $G(k_1, k_2) = 0$, otherwise $G(k_1, k_2) > 0$.

In this paper, "Facility units k_1 and k_2 are adjacent" means that Facility unit k_1 is assigned to one of Cell (*i*-1, *j*), Cell (*i, j*-1), Cell (*i*+1, *j*) and Cell (*i, j*+1), if Facility unit k_2 is assigned to Cell (*i, j*) ($2 \le i \le m-1$ and $2 \le j \le n-1$), as shown in Figure 2. We neglect Cell (*i, j*)s for *I* < 1, *m* < *i* , *j* < 1, or *n* < *j*.

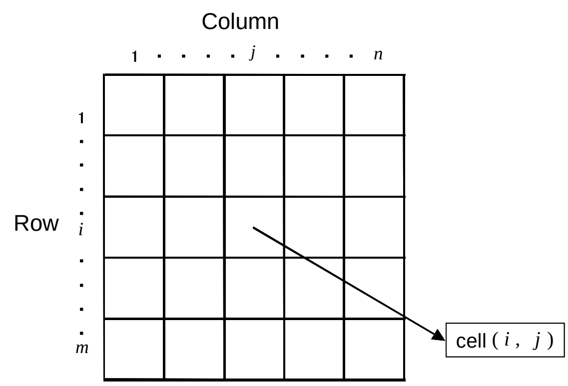

Figure 1. Layout of cells.

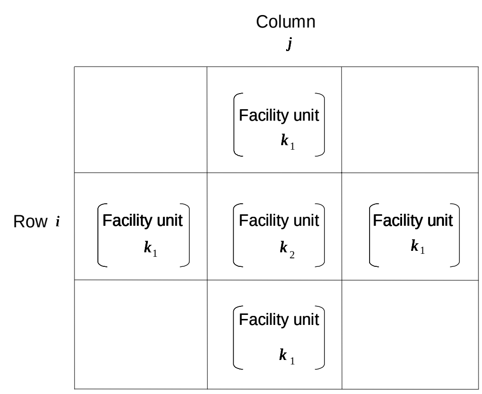

Figure 2. Definition of adjacency.

Note that the layout plan is fixed if *mn* dimensional vector *P* is given. So, we call *mn* dimension vector *P* "layout P" in the latter discussion.

By using the above-mentioned notations, our considered objective function $I(P)$ are expressed as follows.

$$I(P)=\sum_{i_1=1}^{m}\sum_{i_2=1}^{m}\sum_{j_1=1}^{n}\sum_{j_2=1}^{n}\Big(F(p(i_1,\ j_1),p(i_2,\ j_2))D((i_1,\ j_1),(i_2,\ j_2)) + G(p(i_1,\ j_1),p(i_2,\ j_2))\Big) \quad (1)$$

You may change the weights of total material handling cost and total adjacent factor by all material handling costs $F(\cdot)s$ and adjacent factors $G(\cdot)s$ between facility units.

Then, we aim to find the facility layout P^* such that

$$I(P^*)=\min_{P}(I(P))$$

in our considered facility layout problem.

3. THE PROPOSED ALGORITHM

In this section, we explain our proposed algorithm in detail. Our proposed algorithm is based on SA and our original idea is in the neighborhood creating procedure used in SA. First, we describe the neighborhood creating procedure in section 3.1 and then outline of our proposed algorithm in section 3.2.

3.1 Neighborhood creating procedure

First, for $k_1 \neq k_2$, $k_1=1,2,\cdots,mn$, $k_2=1,2,\cdots,mn$, adjacent preference value $v(k_1,k_2)$ is defined as

$$v(k_1,\ k_2)=F(k_1,\ k_2)+F(k_2,\ k_1)+G(k_1,\ k_2)+G(k_2,\ k_1) \quad (12)$$

where k_1 and k_2 are Facility unit numbers.

If $v(k_1,k_2)$ is larger than others, Facility unit k_1 should be assigned into the adjacent cell of Facility unit k_2. This is the local preferable assignment, not global. However, we aim to get the optimal or good layout by the build-up of 2 facility units with large adjacent preference value. By using this adjacent preference value, we describe our proposed neighborhood creating procedure in the following. Now, we get layout P as a temporary layout.

<STEP 1>
Select one cell randomly. We let this cell be Cell (i,j), where $2 \le i \le m-1$ and $2 \le j \le n-1$. Then, note that the number of adjacent facility units is 4.

<STEP 2>
Obtain adjacent preference value of $v(p(i,j),l)$ for $l=1,2,\cdots,mn$ and $l \neq p(i,j)$, where we suppose

$$v(p(i,j),\ l_1) \ge v(p(i,j),\ l_2) \ge v(p(i,j),\ l_3) \ge \cdots \ge v(p(i,j),\ l_{mn-1})$$

That is, l_i is the facility unit number with i-th largest adjacent preference value.

<STEP 3>
Select 4 Facility units l_1,l_2,l_3 and l_4, and we assign these 4 Facility units into 4 cells adjacent to Cell (i,j). After that, we get the objective function value in the new (temporary) layout. As there are 4! ways of assigning these 4 Facility units around Cell (i,j), we have to check the objective function values in all 4! cases.

<STEP 4>
As a current solution, select a layout with the best objective function value out of these 4! values.

Remarks: At STEP1, when a cell is selected except the above, that is, a cell on the edge of rectangle, the cell has only 2 or 3 cells as the neighborhood cells, we can select the current solution with the similar manner.

By using the example as shown in Figures 3 and 4, we demonstrate our proposed neighborhood creating procedure described in the above.

2	12	***14***	5
1	***4***	***6***	***16***
13	3	***11***	8
7	10	15	9

Figure 3. Example of the selected facility unit and four adjacent facility units.

2	12	3	5
1	7	***6***	8
13	***14***	10	***16***
4	***11***	15	9

Figure 4. Example of created neighborhood by using proposed method.

<STEP 1>
At this STEP, we select one cell randomly and the selected cell is supposed to be Cell(2, 3), where Facility unit 6 is assigned as shown in Figure 3.

<STEP 2>
As $p(2,3)=6$, we obtain adjacent preference values of $v(6,l)$ for $l=1, 2, \cdots, 16$ and $l \neq 6$. As a result, we suppose

$v(6,3) > v(6,10) > (6,8) > (6,7) > \cdots$ That is $l_1 = 3$, $l_2 = 10$, $l_3 = 8$ and $l_4 = 7$

<STEP 3>
From the results of STEP 2, Facility units 3, 10, 8 and 7 are selected to be assigned into adjacent cells of Cell(2, 3). In Figure 4, Facility units 4, 11, 14 and 16 at adjacent cells of Cell(2, 3) are exchanged with Facility units 3, 7, 8, 10. By using enumeration of assignment of Facility units 3, 7, 8 and 10 into Cells(1, 3), (2, 2), (2, 4) and (3, 3), 4! new (temporary) layout alternatives can be created as neighborhood of current layout in Figure 3. One of the new layouts is shown in Figure 4.

<STEP 4>
Select the best one of the layouts with the smallest objective function value from 4! layout alternatives created in STEP3 and adopt it as the updated current solution.

3.2 The proposed algorithm

In this section, three algorithms are explained. The first (as "SA algorithm") is the algorithm based on SA (Kirkpatrick *et al.*, 1983) with random 2-opt operation as the neighborhood generation procedure. The second algorithm (as "SA1 algorithm") is also based on SA but uses the procedure described at section 3.1 as the neighborhood generation procedure. The third algorithm (as "SA2 algorithm") is also based on SA, but uses 2 procedures in SA and SA1, which are chosen with a given probability, as the neighborhood generation procedure. These algorithms are explained as follows:

SA algorithm (as shown in Figure 5):
Step1: Initialize the temperature value.
Step2: Create the initial solution (the initial layout) and set it as the current solution.
Step3: If the terminate condition holds (the counter of created solution arrives at the maximum value), output the best solution and terminate this algorithm, otherwise, go to Step4.
Step4: Create neighborhood of the current solution by using random 2-opt operation and select the best neighborhood.
Step5: Determine the acceptance or rejection operation according to acceptance probability. When acceptance, go to Step6, otherwise go to Step7.

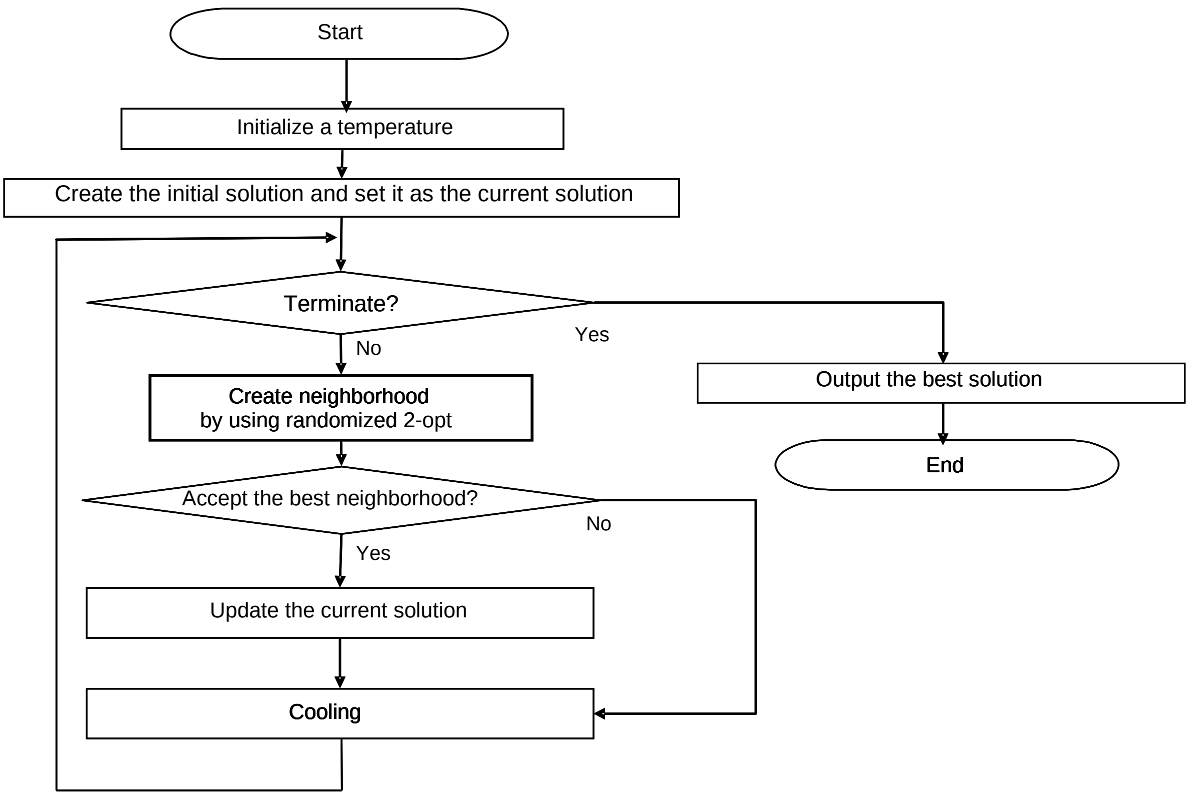

Figure 5. SA algorithm.

Step6: Update the current solution.
Step7: If iteration counter arrives the maximum value of iteration, set new temperature by cooling ratio and reset the iteration counter. Otherwise, returns to Step3.

SA1 and SA2 algorithms are the same as SA algorithm except Step4. Step4 in SA1 and SA2 are described in the following.

SA1 algorithm
Step4: Create neighborhood of the current solution by using Neighborhood generation method described in section 3.1 and select the best neighborhood.

SA2 algorithm
Step4: Create neighborhood of the current solution by adjacent preference (AP) and randomized 2-opt (R-2opt). The AP and R-2opt are chosen by probability and select the best neighborhood.

4. EVALUATIION OF THE ALGORITHM

In this paper, we conducted numerical experiments to evaluate these three algorithms. The specification of computer used in the experiment is as follows.

- CPU: Pentium IV (3.0GHz)
- RAM: 512MB
- Language: Microsoft Visual C++ Version 6.0
- OS: Microsoft Windows XP

4.1 Numeric example

There are two kinds of problem solved in the experiment. One is Problem type 1 (“optimum-known problem”) (Suzuki *et al*., 2004). the other is Problem type 2 (“random-prepared and optimum-unknown problem”) (Suzuki *et al*., 2004).

Problem type 1 is gotten by
1) Deciding one layout
2) Giving material handling costs and adjacent factors to layout according to present layout. Material handling costs between facility units look like network, because there is material flow only between neighboring facility units. So, we call this type of problem network flow data problem.

Problem type 2 is gotten by
1) Giving material handling costs and adjacent factors randomly. Material handling costs and adjacent factors between facility units are uneven, because there is Material handling costs or adjacent factor not only between neighboring facility units in the best layout. So, we call this type of problem type 2.

All of the shape of layout area is square ($m = n$). The numbers of facility units are 64, 81, 100, 121, 144, 169, 196, 225 (but, 121, 144, 169, 196, 225 are only in Problem type 1) as problem sizes. In their problems, we set the material handling cost and the adjacent factor with the following manner. First, we determine the percentage of their values being 0 and we call the percentage sparse degree. The material handling costs and the adjacent factors set to 0 are selected randomly. The material handling costs and the adjacent factors except 0 are set randomly as the integer between 1 and 5. With the above procedure, we made 3 problems per problem size. Each problem was solved five times by each algorithm. To compare with the existing studies, we set sparse degree to 97% when the number of facility unit is 121, 144, 169, 196, 225 and 90% when the number of facility unit is 64, 81, 100.

In this paper, we dealt with Problem type 1 and the Problem type 2 which were explained above. As the evaluation criteria, we used the mean of the objective function value (OFV), which is observed in a given iteration number of calculation.

4.2 Comparison of the result

In this section, we explain the results of experiments. Table 1 shows the mean of OFVs obtained by each algorithm at Problem type 1 and Table 2 shows them at Problem type 2. The underlined values in these tables indicate the best (lowest) one of means of OFVs from comparison of three alogrithms solving one problem. From Table 1, we recognized that SA1 was most effective for solving problems of Problem type 1. On the other hand, SA2 was the most efficient algorithm to solve problems of Problem type 2.

It is shown, in Table 1, that SA1 is effective for solving the problem type 1. In the case of 100 and 121 facility units problems, by using SA1, we could find the optimum solution at all five trials for problems No.1 and No.2. In larger size problems, SA1’s mean of objective function value was smaller than SA’s mean of OFV. These results are because of our proposed neighborhood generation method. Our method enables us to make local optimum layout by introducing adjacent preference and it makes possible to get better OFV.

Table 2 shows that SA2 is good for the problem type 2. This results from SA2 with merit of both of SA1 and SA. In the problem type 2, every facility unit has material handling costs between many facility units, so, it becomes difficult to make local optimum layout, But SA1 algorithm obtains good local layout by making use of adjacent preference in SA2 algorithm and SA algorithm enables us to remove bad local layouts as the current layout. So, we guess we could get better mean of OFV than SA.

Table 1. The comparison of the mean of OFV in the case of solving Problem type 1

mn	Problem No.	Mean of OFV		
		SA	SA1	SA2
100	1	5740	3600	3600
	2	5448	3600	3600
	3	5484	3792	4372
121	1	6444	4400	5360
	2	6032	4400	6756
	3	6072	5664	5716
144	1	8944	5924	6300
	2	7564	6060	6968
	3	8284	6844	7479
169	1	10200	6760	10520
	2	10600	7756	10128
	3	9700	7020	7620
196	1	12576	9168	13052
	2	13860	9108	13552
	3	14176	9308	13480
225	1	16944	12556	16928
	2	16376	11340	15816
	3	16672	10816	16228

Table 2. The comparison of the mean of OFV in the case of solving Problem type 2

mn	Problem No.	Mean of OFV		
		SA	SA1	SA2
64	1	57644	62732	57524
	2	57666	62446	57374
	3	57570	62764	57554
81	1	108438	120218	107774
	2	108566	120428	107804
	3	108692	120636	107828
100	1	183220	215938	183384
	2	183256	199492	182826
	3	183622	199594	182818

5. CONCLUSION

In this paper, we proposed the SA-based algorithms with the new neighborhood generation method by using adjacent preference and randomized 2-opt operation for solving the cell type facility layout problem. Within the results of our conducted experiment, we could get better solution with SA1 algorithm than the existing SA to solve Problem type 1 and with SA2 algorithm than the existing SA to solve Problem type 2 respectively. As the future work, we will develop the efficient algorithm for the problem which the number of facility unit is more than 225. Moreover, the effective algorithms are developed for multi-floor facility layout problem considering material handling cost and adjacent factors.

ACKNOWLEDGMENT

This research was supported in part by the Grant-in-Aid for Scientific Research 18510129 from Japan Society of Promotion of Science (JSPS).

REFERENCES

Burkard, R. E. (1984), Quadratic Assignment Problems, *European Journal of Operational Research*, **15**, 283-289.

Dutta, K. N. and Sahu, S. (1982), A multi goal heuristic for facilities design problem: MUGHAL, *International Journal of Production Research*, **20**(2), 147-154.

Fortenberry, J. C. and Cox, J. F.(1985), Multiple criteria approach to the facilities layout problem, *International Journal of Production Research*, 23(4), 773-782.

Foulds, L. R. (1983), Techniques for Facilities Layout: Deciding Which Pairs of Activities Should Be Adjacent, *Management Science*, **29**(12), 1414-1426.

Golany, B. and Rosenblatt, M. J. (1989), A Heuristic Algorithm for the Quadratic Assignment Formulation to the Plant Layout Problem, *International Journal of Production Research*, **27**(2), 293-308.

Harmonosky, C. M. and Tothero, G. K. (1992), A multi-factor plant layout methodology, *International Journal of Production Research*, **30**(8), 1773-1789.

Houshayar, A. and White, B. (1993), Exact Optimal Solution for Facility Layout: Deciding Which Pairs of Locations Should Be Adjacent, *Computers and Industrial Engineering*, **29**(2), 177-187.

Kirkpatrick, S. *et al.* (1983), Optimization by Simulated Annealing *Science*, **220**(4598), 671-680.

Kon, T., Fukushi, T., Suzuki, A., and Yamamoto, H. (2006), Algorithm for solving facility layout problems by using simulated annealing, *Proceedings of Japan Industrial Management Association 2006 fall meeting*, 160-161(in Japanese).

Koopmans, T. C. and Beckman, M. (1957), Assignment Problems and the Location of Economic Activities, *Econometrica*, **25**, 53-76.

Lacksonen, T. A. and Enscore, E. E. Jr., (1993), Quadratic Assignment Algorithms for the Dynamic

Layout Problem, *International Journal of Production Research*, **31**(3), 503-517.

Rosenblatt, M. J. (1979), The facilities layout problem: a multi-goal approach, *International Journal of Production Research*, **17**(4), 323-332.

Suzuki, A., Yamamoto, H., Nakamura, T., and Tsujimura, Y. (2004), Efficient Algorithms for Solving the Facility Layout Problem by using Symmetries, *the proceeding of Eighth International Conference on Manufacturing and Management*, **2**, 601-608.

Suzuki, A., Yamamoto, H., and Nakamura, T. (2005), Solving Methods for Cellular Facility Layout Problems by using Evolutionary Algorithms, *Journal of Society of Plant Engineers in Japan*, **17**(2), 51-59(in Japanese)

Urban, T. L. (1987), A multiple criteria model for the facilities layout problem, *International Journal of Production Research*, **25**(12), 1805-1812.

Yamamoto, H., A. Suzuki, T. Nakamura, (2004), A Branch and Bound Algorithm for Solving Facility Layout Problems Using Symmetries, *Journal of Japan Industrial Management Association*, **55**(5), 293-301(in Japanese).