

Tree Structure Modeling and Genetic Algorithm-based Approach to Unequal-area Facility Layout Problem

Terushige Honiden[†]

Department of Industrial Management
Osaka Institute of Technology, Osaka 535-8585, JAPAN
Tel: +81-6-6954-3374, E-mail: honiden@dim.oit.ac.jp

Abstract. A tree structure model has been proposed for representing the unequal-area facility layout. Each facility has a different rectangular shape specified by its area and aspect ratio. In this layout problem, based on the assumption that the shop floor has enough space for laying out the facilities, no constraint is considered for a shop floor. Objectives are minimizing total part movement between facilities and total rectangular layout area where all facilities and dead spaces are enclosed. Using the genetic code corresponding to two kinds of information, facility sequence and branching positions in the tree structure model, a genetic algorithm has been applied for finding non-dominated solutions in the two-objective layout problem. We use three kinds of crossover (PMX, OX, CX) for the former part of the chromosome and one-point crossover for the latter part. Two kinds of layout problems have been tested by the proposed method. The results demonstrate that the presented algorithm is able to find good solutions in enough short time.

Keywords: Facility layout, Tree structure, Genetic algorithm.

1. INTRODUCTION

A facility layout problem (FLP) is one of the important areas in a design of production systems. The objective of FLP is to determine the placements for a group of facilities, machines and so on in a shop floor. So many procedures to solve FLP have been proposed previously. FLP can be divided into two categories by the facility area: an equal-area facility layout problem and an unequal-area facility layout problem. The equal-area FLP can be usually modeled as a quadratic assignment problem, for which numerous algorithms have been proposed. But practical layout problems are almost unequal-area FLP. For this problem, Armour and Buffa (1963) proposed an adjacent departments exchange method. Bazaraa (1975) presented a branch and bound approach and van Camp et al. (1991) proposed a nonlinear programming model and a heuristic method. As the facility is usually assumed to be rectangular in unequal-area FLP, it can be specified by its area and aspect ratio. The unequal-area facility problems occur in many environments such as manufacturing cell layout. The area and the shape of cells are different from each

other. The shape of the cell may be changed by re-laying out the machines in the cell. Therefore, a feasible range of aspect ratio should be considered. Tam (1992a) represented the cell layout in a specified floor shop as a slicing tree and developed the algorithm using simulated annealing. Recently the researches using genetic algorithm to the unequal-area FLP have increased (Tam 1992b, Tate and Smith 1995, Rajasekharan et al. 1998). But there are many restrictions on solving the FLP to obtain the optimal layout in a fixed floor space. In this paper, we propose a tree structure model that expresses the layout process, and provide a new genetic algorithm based approach to solve the unequal-area facility problem under the assumption that a shop floor has enough space for the facility layout.

2. UNEQUAL-AREA FACILITY LAYOUT PROBLEM

2.1 Problem description

A facility is assumed to have a rectangular shape and

[†] : Corresponding Author

be specified by its area and aspect ratio. The aspect ratio a_i of the facility i is defined as

$$a_i = \frac{\text{height of rectangular area of facility } i}{\text{width of rectangular area of facility } i} = \frac{h_i}{w_i} \quad (1)$$

We can impose a constraint on the shape of the facility by restricting its aspect ratio to $[a_i^L, a_i^U]$, where a_i^L and a_i^U are the lower and the upper bound of a_i .

The first objective in this facility layout problem should be minimizing of total part transfer cost between facilities. Also, considering that the shop floor area is not specified in this problem, the minimization of total layout area where all facilities and dead spaces are enclosed is set as the second objective.

Hence, the facility layout problem is defined as following two-objective problem:

$$\text{Min } f = \sum_{i=1}^{n-1} \sum_{j=i+1}^n c_{ij} \cdot v_{ij} \cdot d_{ij} \quad (2)$$

$$\text{Min } S = H \cdot W \quad (3)$$

Subject to

$$a_i^L \leq a_i \leq a_i^U, \quad i = 1, 2, \dots, n \quad (4)$$

$$s_i = h_i \times w_i, \quad i = 1, 2, \dots, n \quad (5)$$

Here,

c_{ij} : unit transfer cost between facility i and j ,

v_{ij} : number of part movements between facility i and j ,

n : number of facilities,

d_{ij} : distance between centroid of facility i and j ,

s_i : area of facility i ,

S : total rectangular area that covers all facilities,

H : height of the total layout area,

W : width of the total layout area.

The next level of heading is boldface with upper and lower case letters. The heading is flushed left with the left margin.

2.2 Tree structure modeling

We can represent any layout process of unequal-area facilities by the tree structure model as shown in Figure 1. A shaded area in the layout is a dead space. The numerical numbers in the tree represent the facility numbers and the letter denotes an operation that determines a facility orientation when laying the facility adjacently with another one. A basic orientation of each facility is given previously. The operation “ h ” indicates that the facility is located in the same direction as the original, and the operation “ v ” shows a rotation of the facility in 90° . We may choose the operation so that it produces the minimum centroid distance for two adjacent facilities.

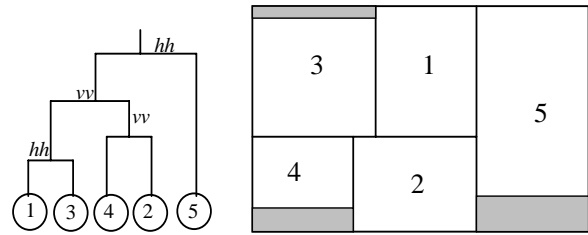


Figure 1. An example of a layout and its tree structure

We can produce various layouts by changing the facility order and the tree structure. To solve the unequal-area facility layout problem is equivalent to the determination of a tree structure corresponding to the layout with a minimum objective function value.

3. GENETIC ALGORITHM APPROACH

We apply a genetic algorithm to solve the unequal-area facility layout problem, that is, to find the optimal tree structure.

3.1 Representation

The tree structure can be represented by the following chromosomes;

$$(a_1 a_2 a_3 \dots a_n : b_1 b_2 b_3 \dots b_{n-2}).$$

The chromosome contains two parts of numerical strings a_i, b_j ; a string of a_i shows the order of the facility in a tree structure, a string of b_j shows the grouping order of facilities. The character “:” is a delimiter. For example, the chromosome corresponding to the tree structure in Figure 2 is as follows;

$$(1 \ 3 \ 4 \ 2 \ 5 : 1 \ 2 \ 1)$$

The order of genes “1 3 4 2 5” is the same as the facility order in Figure 2. The latter genes “1 2 1” is corresponding to the numerical number with “*” in the structure in Figure 2.

The number “1*” shows that the first facility “1” and the next facility “3” are grouped at first. Then “2*” indicates that the second facility “4” and the next facility “2” are grouped at the second grouping procedure. At the third grouping procedure, the first facility group “1 3” and the next facility group “4 2” are grouped. As it is obvious that the last two facility groups are merged, the last number “1*” is omitted in the chromosome. Then the number of the genes in b_j is $(n-2)$. Also, it is obvious from Figure 2 that the value of b_j is limited $1 \leq b_j \leq n-j, j=1, 2, \dots, n-2$.

If two facilities are exchanged in the tree, a pair of facilities combined adjacently is changed. If the tree structure is changed, the facility grouping process is changed. For example, when the tree structure in Figure 2

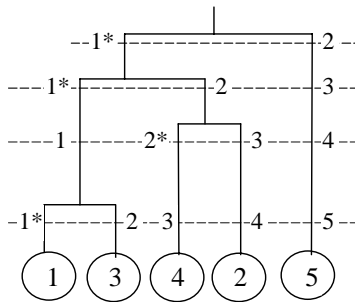


Figure 2. Tree structure

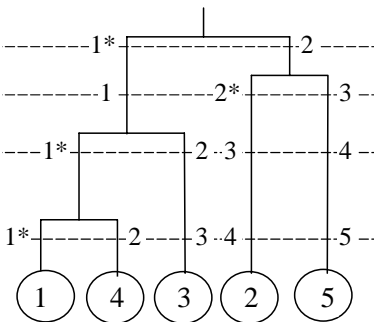


Figure 3. Modified tree structure

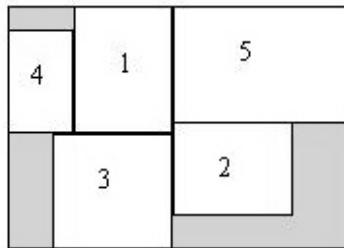


Figure 4. Modified facility layout

is changed to Figure 3, the chromosome can be represented as (1 4 3 2 5 : 1 1 2). And the facility layout corresponding to the modified tree structure is shown in Figure 4. We can generate various kinds of layout by changing the genes in the tree structure chromosome.

3.2 Genetic algorithm procedure

3.2.1 Initialization

The initial population of chromosomes is generated randomly. The genes of the latter part of the chromosome are generated on condition $1 \leq b_j \leq n-j$ described above.

3.2.2 Evaluation

As the facility layout problem defined in 2.1 has two

objectives, non-dominated solutions may be obtained. It is difficult to evaluate the non-dominated solutions. Then, we define the following single objective function to evaluate a chromosome. The meaning of z_i is shown in Figure 5.

$$z_i = \frac{f_i}{f_{max}} \cos \theta + \frac{S_i}{S_{max}} \sin \theta \tag{6}$$

Here,

f_i : total part transfer cost for chromosome i ,

S_i : total layout area for chromosome i ,

f_{max} : maximum value of f_i ,

S_{max} : maximum value of s_i ,

θ : random value between 0 and $\pi/2$.

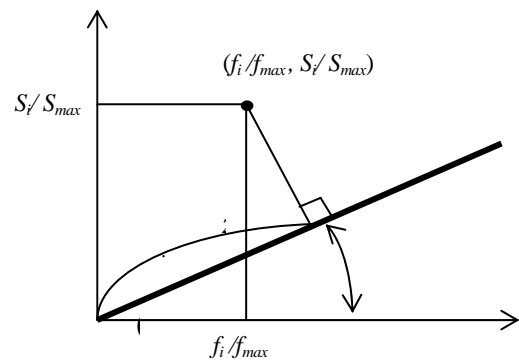


Figure 5. Evaluation of chromosome

Fitness function for the solution corresponding to chromosome i is defined as follows;

$$F_i = z_{max} - z_i \tag{7}$$

Here, $z_{max} = \cos \theta + \sin \theta$. z_{max} is the value of z_i when $f_i = f_{max}$, $s_i = s_{max}$. This fitness function is used in a selection procedure.

When we select the chromosome for the next generation, non-dominated chromosomes are chosen at first. If the number of non-dominated chromosomes is less than the population size, dominated chromosomes are added to the population.

3.2.3 Crossover

For the genes of the former part of a chromosome, PMX(Partially Mapped Crossover), OX(Order Crossover) and CX(Cycle Crossover) are used. These crossovers do not generate infeasible offspring. One-point crossover is used for the latter part of genes.

3.2.4 Mutation

The mutation is carried out by exchanging two genes selected in the former part of offspring randomly. The mutation for the latter part is that the value of one gene b_j

selected randomly is changed on condition $1 \leq b_j \leq n-j$.

3.2.5 Selection

A pair of chromosomes is picked from the population using the roulette wheel selection. The probability of selecting the chromosome i is calculated as follows;

$$P_i = \frac{F_i}{\sum_{j=1}^{PS} F_j} \tag{8}$$

Here PS is a population size. When calculating z_i in F_i at each generation, θ is given randomly. Then we can select various non-dominated chromosomes.

4. NUMERICAL EXAMPLES

In order to demonstrate the efficiency of the proposed GA method, we apply it to two kinds of layout problems; 10-facility layout problem and 20-facility layout problem.

(1) 10-facility layout problem

The basic data of area, aspect ratio for each facility and part movement are shown in Table 1 and Table 2 (from Tam and Li 1991). The unit transfer cost c_{ij} is set 1. The genetic parameters are set as follows;

Population size=30, crossover rate=0.5,
mutation rate=0.2, maximum generation=500.

We run the genetic algorithm 5 times and compare the GA solutions with the solutions obtained by setting tree structure data randomly. As we cannot know true non-dominated solutions for this problem, we use the best solutions of all obtained by GA and random procedure as the true non-dominated solutions.

Figure 6 indicates the random solutions and GA solutions for three kinds of crossover operators and the best non-dominated solutions. It is obvious that the genetic algorithm approach provides good solutions. The computation time for one trial (500 generations) is about 3.5 minutes.

To evaluate the non-dominated solutions obtained by GA, we use two kinds of measures; the degree of point approximation (DPA) and the degree of set approximation (DSA) (Morizawa 1995).

DPA is a measure of the closeness of a solution to the nearest true non-dominated solution. As shown in Figure 7, DPA_i for non-dominated solution P_i is defined as follows.

$$DPA_i = \text{Min}_{k=1, \dots, n_K} \frac{D(P_i, P_{NDk})}{D(P_W, P_{NDk})} \tag{9}$$

Here,

$P_i = (f_i, s_i)$; non-dominated solution obtained by genetic algorithm approach, $i=1,2,\dots,n$,

$P_{NDk} = (f_{NDk}, s_{NDk})$; the k th true non-dominated solution, $k=1,2,\dots,n_K$,

$P_W = (f_{max}, s_{max})$; the worst solution in the obtained solutions, $D(p1, p2)$; Euclidean distance between $p1$ and $p2$

$$D(p_1, p_2) = \sqrt{\left(\frac{f_{p1} - f_{p2}}{f_{max}}\right)^2 + \left(\frac{s_{p1} - s_{p1}}{s_{max}}\right)^2} \tag{10}$$

Table 1. Cell area and aspect ratio

Cell No	Area	Aspect ratio	
		Lower	Upper
1	24.0	0.80	1.00
2	16.0	0.75	1.15
3	36.0	0.60	1.85
4	8.0	0.30	1.10
5	21.0	0.90	1.18
6	17.5	0.50	1.00
7	3.6	0.30	1.40
8	15.4	0.60	1.25
9	20.0	0.90	1.00
10	19.5	1.20	1.80

Table 2. The number of part movement v_{ij}

$i \backslash j$	1	2	3	4	5	6	7	8	9	10
1	-	10	0	5	1	0	1	2	2	2
2		-	1	3	2	2	2	3	2	0
3			-	10	2	0	2	5	4	5
4				-	1	1	5	0	0	2
5					-	3	5	5	5	1
6						-	2	2	1	5
7							-	6	0	1
8								-	5	2
9									-	0
10										-

As shown in Figure 8, DSA means the difference between the dominated area specified by the obtained solutions and the true dominated area. DSA is defined as follow;

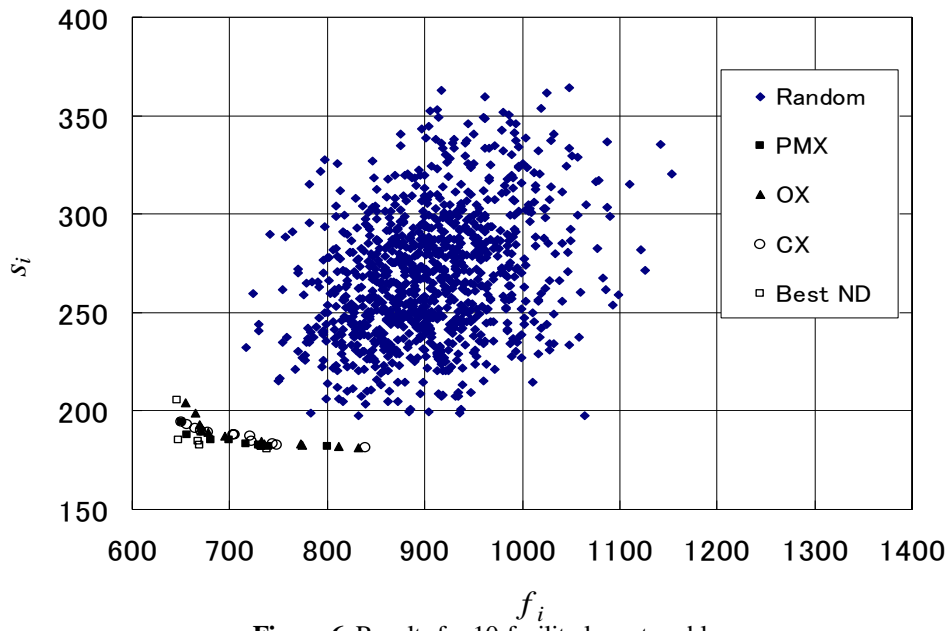


Figure 6. Results for 10-facility layout problem

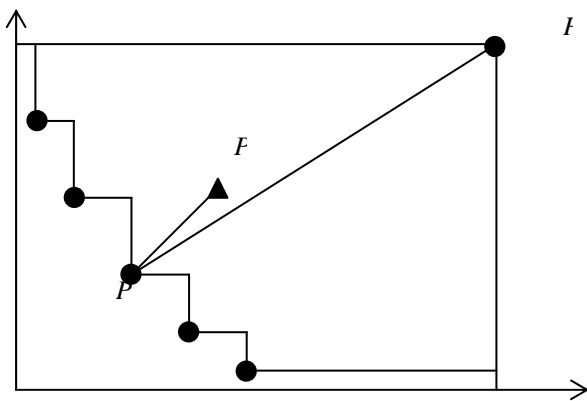


Figure 7. Degree of point approximation

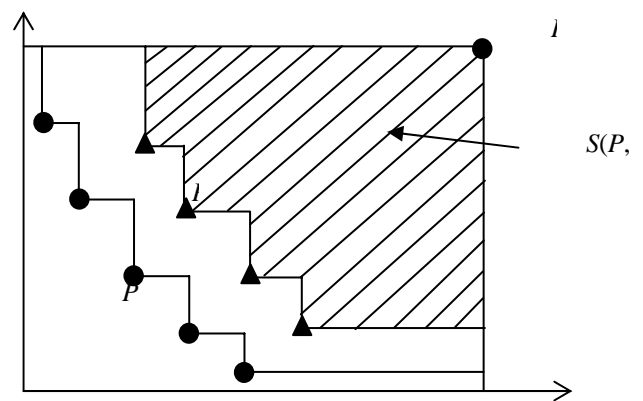


Figure 8. Degree of set approximation

$$DSA = \frac{S(P_{ND}, P_W) - S(P, P_W)}{S(P_{ND}, P_W)} \quad (11)$$

Here,

$S(P_{ND}, P_W)$; Area enclosed by $P_{NDk}, k=1,2,\dots,n_K$ and P_W ,

$S(P, P_W)$; Area enclosed by $P_i, i=1,2,\dots,n$ and P_W .

Table 3 shows the evaluation value of DPA and DSA for GA solutions and the random solution. DPA and DSA of GA solutions are very small rather than the random solution. From Table 3, PMX operation has the minimum value of DPA and DSA .

Table 3. Evaluation of GA solution and random solution

		DPA	DSA
GA	PMX	0.022	0.036
	OX	0.024	0.043
	CX	0.030	0.041
RANDOM		0.095	0.191

(2) 20-facility layout problem

We apply GA method to the 20-facility layout problem cited from Imam and Mir (1993). The GA parameters are set as follows;

Population size=40, crossover rate=0.5,
mutation rate=0.1, maximum generation=500.

Table 4 shows the non-dominated solutions by GA approach with PMX operation and the solution by Imam and Mir. As they solved the problem as a single objective problem, the total layout area of their solution is large. But by GA approach, we obtained three non-dominated solutions with small layout area.

Table 4. Solutions of 20-facility layout problem

	f	S
Imam and Mir	1264.94	154
GA	1276.00	132
	1283.50	120
	1313.01	108

5. CONCLUSIONS

In this paper, a genetic algorithm approach is proposed for the unequal-area facility layout problem. We represent a facility layout process using a tree structure model and adopt the tree structure as a chromosome to use in a genetic algorithm. We apply the proposed GA method to two kinds of layout problems and show that the method with PMX

operation is effective to obtain non-dominated solutions.

REFERENCES

- Armour, G. C. and Buffa, E. S., (1963) A Heuristic Algorithm and Simulation Approach to Relative Location of Facilities. *Management Science*, **6**, 294-309.
- Bazaraa, M. S., (1975) Computerized Layout Design: A Branch and Bound Approach", *AIIE Transactions*, **7**, 432-438,
- Imam M.H. and Mir M., (1993) Automated layout of facilities of unequal. *Computer Ind. Eng.*, **24**, 335-366.
- Morizawa K., (1995) A multi-objective scheduling method using complex random sampling. *Doctoral dissertation*, Osaka Prefecture University.
- Rajasekharan, M., Peters, B.A. and Yang, T., (1998) A Genetic Algorithm for Facility Layout Design in Flexible Manufacturing Systems. *Int. J. of Prod. Res.*, **36**, 95-110.
- Tam, K. T. and Li, S. G., (1991) A Hierarchical Approach to the Facility Layout Problem. *Int. J. of Prod. Res.*, **29**, 165-184.
- Tam, K. Y., (1992a) A Simulated Annealing Algorithm for Allocating Space to Manufacturing Cells, *Int. J. of Prod. Res.*, **30**, 63-87.
- Tam, K. Y., (1992b) Genetic Algorithm, Function Optimization and Facility Layout Design. *European J. of Oper. Res.*, **63**, 322-346.
- Tate, D. M.. and Smith, A. E., (1995) Unequal-area facility layout by Genetic Search. *IIE Transaction*, **27**, 465-472.
- van Camp, D. J., Carter, M. W., and Vannelli, A., (1991) A Nonlinear Optimization Approach for Solving Facility Layout Problems. *European J. of Oper. Res.*, **57**, 174-189.