e-ISSN : 0975-3397
p-ISSN : 2229-5631

A Sequence-pair based heuristic for Multi Floor Facility Layout Problem

Shunichi OHMORI, and Kazuho YOSHIMOTO
Department of Industrial and Management System Engineering,
Waseda University, Tokyo, Japan,
e-mail: ohmori0406@aoni.waseda.jp.

***Abstract*—We consider the problem of finding the most efficientlayout of department that are to be placed in the multi-floorfacility. It is known that the Facility Layout Problem (FLP)is a class of NP-hard problem, and thus, is difficult to solve.Multi-Floor Facility Problem, in fact, is much more difficult thansingle-floor problem due to the large number of combinationof binary variables that specify on which floor to assign eachdepartments. Thus, most of the algorithm cannot be applied tothe large size problem where there are typically more than 30departments. In this paper, we proposed a heuristics method thatcan be applicable to such large sized problem. We developeda heuristic inspired by SEQUENCE, that outperformed manyother algorithms in single-floor FLP and expand it to multifloorproblem. In the computational experiments, we applied ourmethod to the problem with 44 departments and showed howthis problem can be solved by our algorithm efficiently.**

***Index Terms*—**Facility Layout Problem; Meta-heuristics

I. Introduction

The Facility Layout Problem (FLP) is the problem of finding the locations of departments with minimal material handling cost represented by the product of material flow times distances between departments.

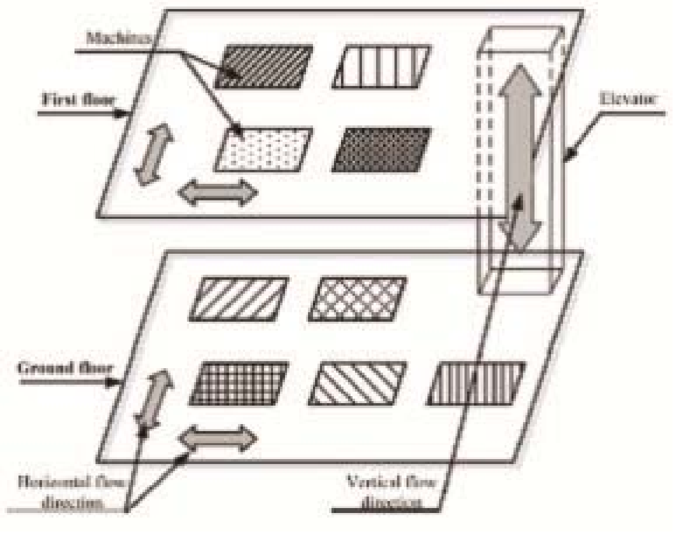

Fig. 1. Multi-Floor Facility Layout Problem [2]

In several papers [3], [4], [5], the vertical movements through the ‘multi-floor departments’ that spread over the floors, such as elevators, or automated warehouses, were considered. In addition, many ‘realistic’ factors have been taken into consideration as: the determination of the number of elevators [?]; the determination of the number of floors and the dimension of facilities [6], I/O points and aisle structures [7].

A solution method for a class of multi-floor FLP has been proposed in several papers. [3] proposed the MULTIPLE, which is the SFC (Space Filling Curves) based techniques for the QAP (Quadratic Assignment Problem) with multi-floor. [4]proposed SABLE, which extends the MULTIPLE by utilizingSA (Simulated Annealing). These approaches, however, usesthe ‘discrete representation’, *i:e:* QAP formulation, and thuseliminate many feasible layouts from solution candidates. Thealternative representation is ‘continuous representation’, whichuses a set of continuous variables to denote the departmentlocations and dimensions. By representing FLP in a continuousmanner, there is no such compromise in solution space.However, solving problem become much more challenging. [9]proposed efficient MIP (Mixed-Integer Programming) basedformulations for multi-floor FLP, which uses a continuousrepresentationof layout. Their formulations include variousacceleration techniques such as symmetry-breaking constraintsand valid inequalities. Even with these acceleration, however,largest problem that can be solved to near optimality contains15 departments, and thus, is not applicable to industrialapplications where there are more than 30 departments.

In this paper, we build an efficient mathematical formulationfor the continuous representation multi-floor FLP. Our formulationis inspired by SEQUENCE [8], which outperformedmany other heuristics in single-floor FLP. We extent thisformulation into multi-floor FLP. We reduce some of theredundant constraints to get more speeding up the process.The outline of the proposed method is the following twosteps: First, the relative positions of each departments arespecified using MFSP (Multi-Floor Sequence Pair), whichis the extension of SP into multi floor layout; Then, thedepartment locations and dimensions are optimized under theabove-specified relative positioning.

We will also consider methods for designing optimal aislestructure for a given layout from our model, using graphtheory-based method. We first create graph, *i:e:* nodes andarcs, onto the obtained layout using 'Min-Max method'. Wethen give aisles for shortest path between I/O (Input/Output)points of each pair of departments, using Dijkstra method. Finally, we will give numerical results and we show how multi-floor layout problem can be solved very efficiently.

II. LOCATION DESIGN

A. Formulation

In this section, we describe our basic formulation for themulti-floor FLP via MIP, *i:e:*, continuous representation.

1) Input Information:

- N : the number of departments
- i, j : department indices ($i, j = 1,...,N$)
- F_{ij}: material flow between departments $i$and j
- s_i: quarter area of department i
- a_i^{min} : the lower limit of the department $i$area
- a_i^{max} : the upper limit of the department $i$area
- g_i: loading of the department i
- G_r: total loading of floor r
- h_i: half of height of department i
- H_r= $\max_r(h_r)$: height of floor r
- R_i:the number of floors spread
- R : the maximum number of floors
- C_S: unit cost of land area
- C_W: unit cost of wall area
- C_G: unit cost of floor reinforcement
- α : parameters for evaluation function

2) Output Information:

- x_i: x-coordinate of center of department i.
- y_i: y-coordinate of center of department i.
- z_i: z-coordinate of center of department i.
- w_i: half of width of department i.
- l_i: half of length of department i.
- W : width of building
- L : length of building
- H: total height of building ($\sum_{r=1}^{R} H_r$)
- G : total loading onto building ($\sum_{r=1}^{R} G_r$).
- a_{ri}= 1 : if department i is assigned to floor r, 0:otherwise.
- X_{ij}= 1 : if department $i$is left of department j, 0:otherwise.
- Y_{ij}= 1 : if department $i$is front of department j, 0:otherwise.
- Z_{ij}= 1 : if department $i$is below department j, 0:otherwise.

3) Objective Function and constraints:

$$\textbf{min.} \quad \alpha C_I + (1-\alpha) C_O \tag{1}$$

$$C_I = C_S WL + C_W(W+L)H + C_G G \tag{2}$$

$$C_O = \sum_{i=1}^{N}\sum_{j=1}^{N}(D_{ij}^H F_{ij}^H + D_{ij}^V F_{ij}^V) \tag{3}$$

$$\textbf{s.t.} \quad w_i \le x_i \le (W - w_i) \tag{4}$$

$$l_i \le y_i \le (L - l_i) \tag{5}$$

$$h_i \le z_i \le (H - h_i) \tag{6}$$

$$a_i^{min} \le h_i / w_i \le a_i^{max} \tag{7}$$

$$w_i l_i \ge s_i \tag{8}$$

$$(w_i + w_j) - (x_i - x_j) - (1 - X_{ij})W \le 0 \tag{9}$$

$$(l_i + l_j) - (y_i - y_j) - (1 - Y_{ij})L \le 0 \tag{10}$$

$$(h_i + h_j) - (z_i - z_j) - (1 - Z_{ij})H \le 0 \tag{11}$$

The objective is to minimize the total sum of initial cost and operational cost.

Initial cost is composed of the total land area cost, wall building cost, and floor reinforcement cost, respectively. Floor reinforcement cost is measured by the sum of the loadings of floors Gr with $r = 1; \_\ \_\ \_ ; R$, where Gr is measured by sum of the loading of departments assigned to the floors above floor r, expressed in equation (12).

$$G_r = \sum_{r'=r+1}^{R}\sum_{i=1}^{N} a_{r'i} g_i \tag{12}$$

Operational cost is material handling cost, that is measured by the material flows and distances between departments in the horizontal and vertical direction. Distances are measured by the rectlinear distances.

The constraints (4)-(7) are within-boundary constraints thatensure that all departments must remain inside the site boundary.The constrains (8) ensures that the area of each departmentremains upper and lower limits. The constraints (9)-(11) arenon-overlapping constraints. These constraints require that thedistances between two departments in x (or y or z) directionsmust be greater than half of sum of their widths (or heightsor length).

B. Sequence-Pair

SP is originally proposed for a related problem in VeryLarge Scale Integration (VLSI) circuit design. A sequencepairconsist of a pair of entities sequences $(\Gamma_+; \Gamma_-)$ thatcan be used to determine the relative locations of entities ina two-dimensional compact space. The relative locations ofdepartment $i$and j for given permutation ($(\Gamma_+; \Gamma_-)$) is specifiedas follows.

- $(\Gamma+;\Gamma-) = (i, j; i, j) \rightarrow i$ is left of j $(X_{ij} = 1)$
- $(\Gamma+;\Gamma-) = (j, i; j, i) \rightarrow i$ is right of j $(X_{ji} = 1)$
- $(\Gamma+;\Gamma-) = (j, i; i, j) \rightarrow i$ is below j $(Y_{ij} = 1)$
- $(\Gamma+;\Gamma-) = (i, j; j, i) \rightarrow i$ is above j $(Y_{ji} = 1)$

There are two advantages in computing cost by using this representation. First advantage is that SP is able to avoid to create the relative-positioning that violates‘transitivity’, *i:e:* if $X_{ij}= 1$ and $X_{jk}= 1$, then $X_{ik}= 1$. Eliminating these infeasible sets of relative-positioning from consideration give a significant impact in computing efficiency. Next advantage is that the total number of combinations that needs to besearched exhaustively with SP is much less than that of the binary variables set $(X_{ij} ; Y_{ij})$. With SP representation, the total number of the combination is calculated as $N!$ X$N!$; whereas, with $(X_{ij} ; Y_{ij})$, the number grows as $2^{2(N \times N)}$. This difference does not seem obvious, but it is surprisingly large. SP-representation-based algorithm is shown to be more effective than other types of heuristics in benchmark problems in [8] as in table 1.

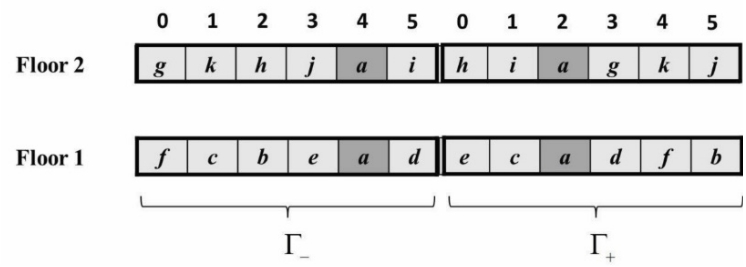

Fig. 2. An example illustrating MFSP for 2-floor MFLP with 10 departments.

TABLE I. NUMERICAL RESULT OF HEURISTICS COMPARISON TEST OF MULTIPLE, SABLE AND SEQUENCE

Problem	MULTIPLE	SABLE	SEQUENCE	Imp.(%)
BM9	252	252	246*	2
M11	1,344	1373	1171*	13
BM12	149	149	142*	4
M15	32,359	31936	28526*	11
M25	1,596	1588	1371*	34
SC30	5,605	6175	3707*	34
SC35	6,086	6733	3604*	41

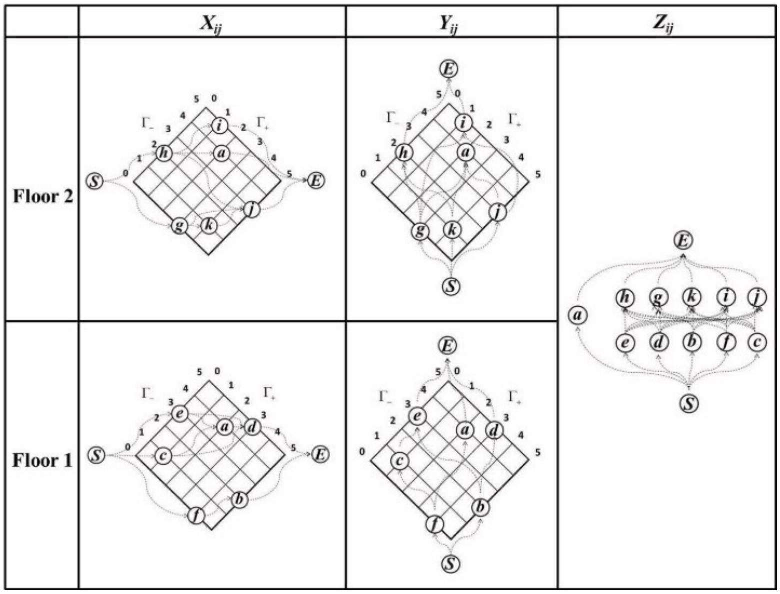

Fig. 3. An example illustrating MFSP for 2-floor MFLP with 10 departments.

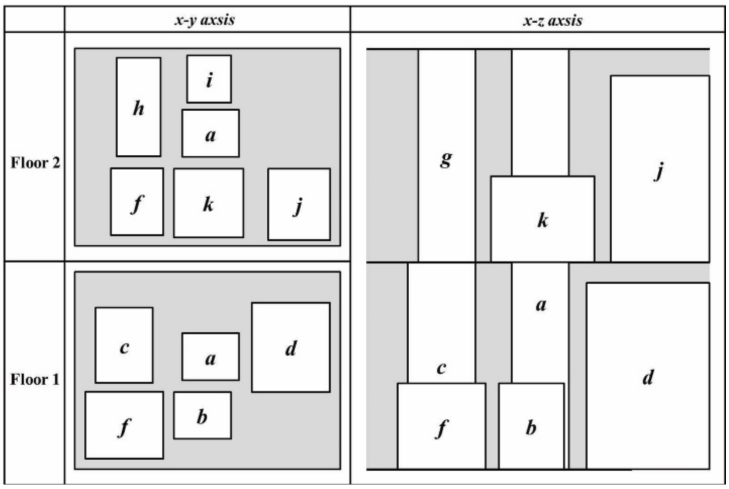

Fig. 4. An optimal layout under specified relative-positioning by MFSP in figure 3.

C. Multi-Floor Sequence-Pair

In this section, we describe MFSP, which extend SP to specify the relative positioning of departments for multi-floor FLP. In MFSP, we have the permutation $(\Gamma_r^+;\Gamma_r^-)$ for each floor *r*. The multi-floor department, e.g. elevators or automated warehouse, are placed in the same location in the permutation for each floor. This proposal is prevent multi-floor department from being disrupted each floor.

An example that contains department *a; b;* _ _ _ *; j* is described In Figure 2-4. Department a is the multi-floor department. Figure 2 expresses $(\Gamma_r^+;\Gamma_r^-)$, and figure 3 expresses relative position imposed by $(\Gamma_r^+;\Gamma_r^-)$, and figure 5 presents the obtained layout for them.

D. Reducing Redundant Relative Constraints

In this section, we describe the redundant inequalities that can be eliminated by exploiting relative positioning structure. Firstelimination is using transitivity property, that is, if the conditions X_{ij}= 1 and X_{jk}= 1 holds, the condition X_{ik}= 1 must hold as well. Therefore, we can reduce the redundant inequalities the non-overlapping constraint for the department *I* and *j*. In the example of figure 3, the X_{ed}or $X_{cd}$are redundant because of $X_{ea};X_{ca};X_{ad}$. Exploiting this property, they can be eliminated to a minimal set of conditions.

Similarly, the relative-positioning only needs to be imposed between the highest department in floor *r* and all departments in floor *r* + 1. Therefore they can be eliminated to a minimal set of conditions as well.

Finally, within-boundary constraints also needs to be imposed for the root nodes (*b; g; e; c; f* in *x* direction and*g; k; j; f; b* in *y* direction for figure 3) and the leaf nodes (*d; b; i; a; j* in *x* direction and *h; i; e; a; d* in *y* direction for figure 3).

E. Optimizing Department Location on Relative Positioning Constraints

By specifying relative locations X_{ij} ;Y_{ij} ;Z_{ij}, the problem (1)-(11) can be reduced to a convex optimization problem, and thus, can be solved to optimality by using standard convex optimization techniques, such as interior-point method. In this paper, we use the primal-dual interior-point method to compute the optimal location under specified relative location via MFSP. Those relative locations can be updated in the SA(Simulated Annealing) process described in the chapter 4.

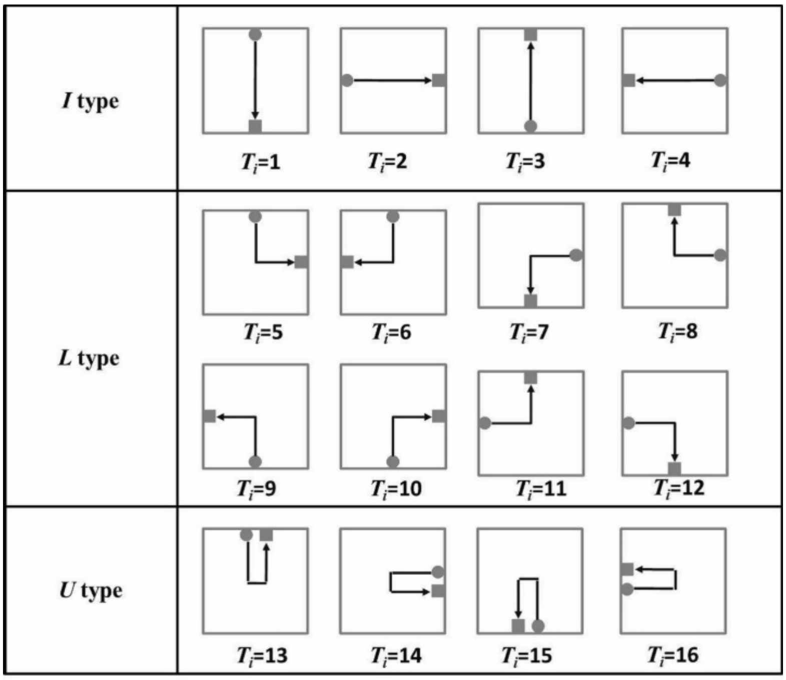

Fig. 5. I/O points type

III. AISLE DESIGN

A. Specifying I/O Point Types

In first step, we specified I/O point types for each department,that are described in the figure 5. we have 16 types ofI/O point, each of which can be grouped into 3 types; ”I”,”U”,or ”L” type. I/O point type for each department is representedby T_i.

These I/O points are used as origin and destination point tomeasure the distances between departments.

B. Generating Nodes using Min Max Method

After specifying I/O point types for each department, wegenerate nodes and arcs on the layout design, using Max-Minmethod. With Max Min method, the nodes are generated forthe next four points. (Example in Figure 6.)

1) Vertex of each department
2) I/O points of each department
3) Max-Min coordinate of each department
4) Intersection of the lines $P_{im}$and $P_{jn}$with $i \neq j$ whereP_{im}denotes the plane passing through previouslycreatednode *i*, *i:e:* vertex, I/O points, and Max-Mincoordinates of each department, in the *m* direction.

After generating nodes, we then create arcs for each pair ofnodes next to each other.

In Figure 6, the case with the department *a; b; c* and *d* isexpressed. (Black points express I/O point of *a* and *d*.)

C. Generating Aisles for Shortest Path between Departments

After creating nodes and arcs, we give aisles onto the arcsthat can be used for the shortest path between each pair ofdepartments. The shortest path between department *i*and *j* iscalculated from Output point of department *i*to Input pointof department *j*, using Dijkstra method.

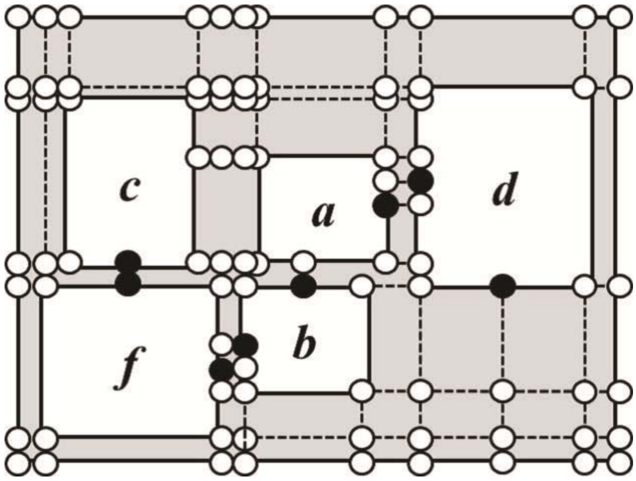

Fig. 6. Nodes and Arcs created onto 1st-floor layout in figure 4

Dijkstra method is an algorithm used in the graph theory to solve shortest path problem. It is invented by Edger Dijkstra in the 1959, and it is the algorithm that seek the shortest path of two apex efficiently.

By using these steps, we can create the ailse structure efficiently.

IV. OVERALL ALGORITHM

In this chapter, we show how to optimize location and aisle design, imposed by a set of $(\Gamma_{+}^{r}, \Gamma_{-}^{r}, T_i)$ as in the form

$$\textbf{min.} \quad (\Gamma_{+}^{r}, \Gamma_{-}^{r}, T_i) = \alpha C_I + (1-\alpha) C_O \qquad (1)'$$

$$\textbf{s.t.} \quad (4)-(11)$$

The problem (1)’,(4)-(11) is a combinational optimization problem, and thus, can be solved via Meta-Heuristics. We use Simulated Annealing (SA) to find an optimal set of $(\Gamma_{+}^{r}, \Gamma_{-}^{r}, T_i)$. The following is a detailed description of the overall algorithm.

A. Simulated Annealing

SA is one of the meta-heuristics proposed in 1983, thename of which come from annealing process in metallurgy.It finds a sequence of points $r_0, r_1, \ldots,$with $f(r_k) \to p^*$as $k \to \infty$, where *k* denote the iteration counts, $r_k$denotethe solution candidate after the *k* th iteration, *f* denotes theobjective function, p^*denotes the optimal solution.

SA’s major advantage over other algorithms is an ability toavoid the convergence at local optimum. The algorithm acceptsnot only changes with the better objective function, but alsosome changes with the worse objective function stochastically.The acceptance probability of worse changes is controlledby the internal parameter, called temperature. The higher thetemperature is, the higher the acceptance probability is. Inmost cases, the temperature is gradually decreased, so that itcan perform more global searches at the beginning, and morelocal searches at the end.

B. Neighborhood Solution

We proposed the following two types of neighborhoodsolution.

1) Swap elements *i*and *j* with $i \neq j$ in $(\Gamma_{+}^{r}; \Gamma_{-}^{r})$
   (*i; j* are randomly chosen)
2) Change the IO point type T_i
   (*i*is randomly chosen)

The first neighborhood is described in figure *, that includethe swap of department within and across floors.

C. Alogrithm Description

Given: $k = 0; r_0; T_0;$
Repeat:(i) to (iv)
(i)Create the new set of candidatesolution.
(ii)Calculate the delta-energy:
(iii)Judge acceptance of new solution
candidate:
(iv)Cool temperature: $T_{k+1} = \gamma T_k$
Until $T_k < T_e$

V. NUMERICAL EXPERIMENT

A. Experimental Outline

To illustrate the effectiveness of the tool, we applied ourmethod to a test problem with 44-department, the data ofwhich comes from a factory for medicinal chemical manufacturingin Japan. The outline of this experiment is explainedbelow.

- Area of department: table 3.
- Material flow between departments: table 4.
- Upper and lower limit of aspect ratio: [0.25,4]
- SA Parameters: $T_0 = 1000, T_e = 100, \gamma = 0{:}9$.

The program is written in MATLAB, the computer specsare Intel (R) Core 2 Duo CPU E6850, 3.00GHz, 3.00GHz,with 4.00GB memory 4.

B. Result of Experiment

The layout design given from the proposed method is shown in Figure 8, and the objective function value and its breakdown are shown in Table II.

As a comment for the obtained layout, we note that there seems a greater emphasis on the material handling efficiency, due to the parameter settings α= 0:3 in this experiment. There are some 'cluster', for example, departments 2,6-10,27,28,35 on floor 2, resulting in considerable operational cost reduction.

On the other hand, there seems some compromises in the investment cost reduction. There are some 'dead spaces' on floor 2 and 3, which result in higher land area cost. Also, heavier departments, such as 2,3,9 (g_i= 14; 28, and 15 respectively) are placed on floor 2 or 3, which result in higher beam reinforcement cost.

Those compromises are considered the result of emphasis on the material handling efficiency. Therefore, our algorithm seems able to reflect 'trade-off' between investment and operation cost properly.

TABLE II. RESULT OF NUMERICAL EXPERIMENT (OBJECTIVE FUNCTION VALUE AND ITS BREAKDOWN)

Item	Value	Unit Cost	Cost
Land area used	269.0	5	1345.2
Wall building	298.6	3	895.8
Beam reinforcement	1141.8	2	2283.7
Material handling	3,814.0	2	7628.1
Total		(α=0.3)	6697.1

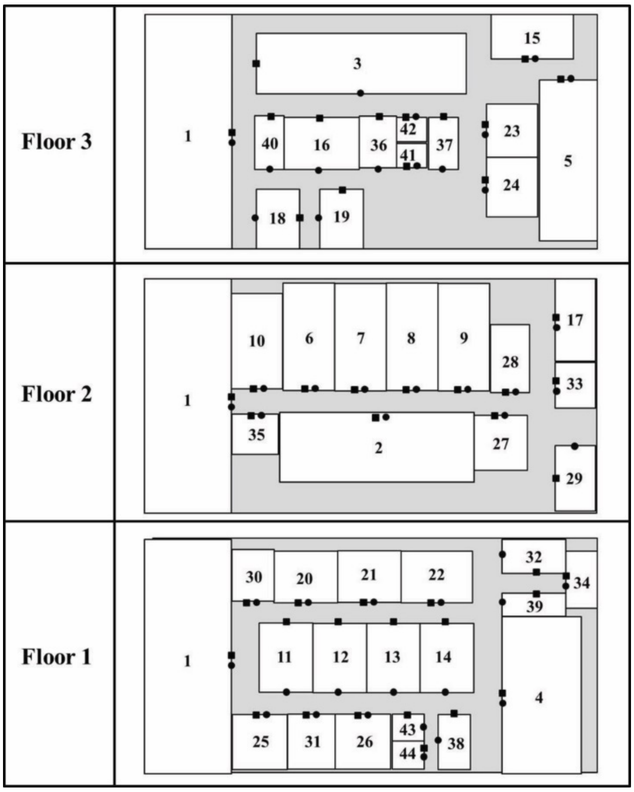

Fig. 8. Layout and Aisle structure obtained from numerical experiment(blackcircles denote Input point and black squares denote Output point).

VI. CONCLUSION

In this paper, we have considered a multi floor facility layoutproblem that is formulated as a Mixed-Integer Programming.We have developed an efficient method inspired by sequencepairrepresentation. In numerical experiment, we showed 44department problem can be solved efficiently by our proposedalgorithm.

REFERENCES

[1] R.V. Johnson, SPACECRAFT for multi-floor layout planning, ManagementSciences, vol.28, no.4, pp.407-417. 1982
[2] A. Drira, H. Pierreval and S. Hajri-Gabouj, Facility layout problems: Asurvey, Annual Reviews in Control, vol.31, no.2, pp.255-267, 2007
[3] Y.A. Bozer, R.D. Meller and S.J. Erlebacher, An improvement-type layoutalgorithm for single and multiple floor facilities, Management Science,vol.40, no.7, pp918-932, 1994
[4] R.D. Meller and Y.A. Bozer, A new simulated annealing algorithm forthe facility layout problem, International Journal of Production Research,vol.34, no.6, pp.1675-1692, 1996
[5] R.D. Meller and Y.A. Bozer, Alternative approaches to solve the multifloorfacility layout problem, Journal of Manufacturing Systems, vol.16,no.3, pp.192-203, 1997
[6] K.Y. Lee, M.I. Roh and H.S. JeongAn improved genetic algorithmfor multi-floor facility layout problems having inner structure walls andpassages, Computers and Operations Research, vol.32, no.4, pp.879-899,2005
[7] D.I. Patsiatzis, L.G. Papageorgiou, Optimal multi-floor process plantlayout, Computers and Chemical Engineering, vol.26 no.5, pp.575-583,2002
[8] Q. Liu and R.D. Meller, A sequence-pair representation and MIPmodel-based heuristic for the facility layout problem withrectangular departments,IIE transactions, vol.39 no.4, pp.377-394, 2007
[9] M. Goetschalckx and T. Irohara, Efficient Formulations for the Multi-Floor Facility Layout Problem with Elevators, Optimization Online, pp.1-23, 2007